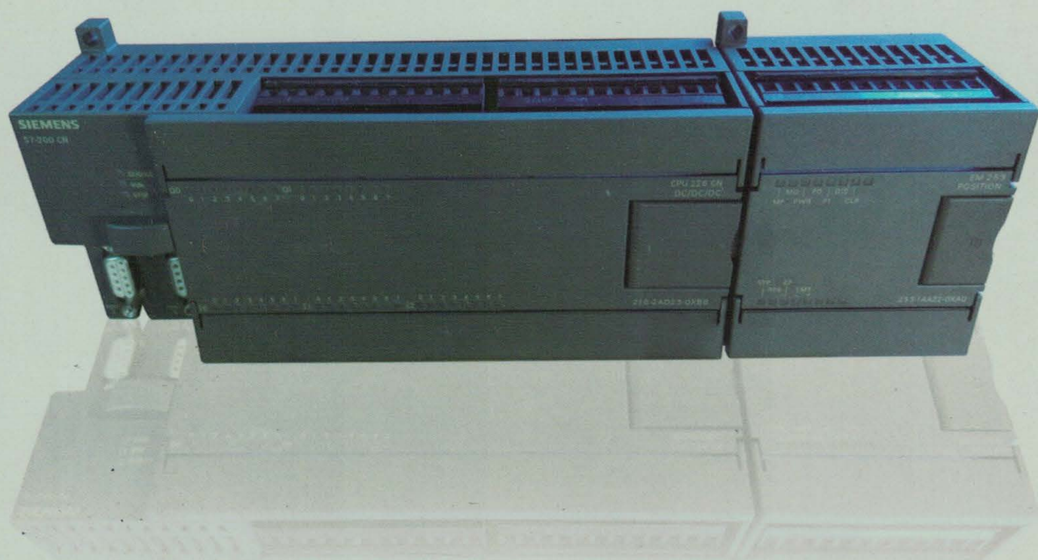
电 气 信 息 工 程 丛 书

S7-200 PLC 基础及工程应用

主编 向晓汉





本书配有源代码和视频资源

下载网址为 www.cmpbook.com





机械工业出版社

CHINA MACHINE PRESS

VISIT…

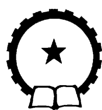
LANZAROTE

Caliente.COM

电气信息工程丛书

S7－200 PLC基础及工程应用

主　编　向晓汉
主　审　郑贞平



机 械 工 业 出 版 社

本书从基础和实用出发，涵盖的主要内容包括 S7－200 系列 PLC 入门、PLC 通信和变频器。全书分两个部分，第一部分为基础部分，主要介绍 S7－200 系列 PLC 的硬件和接线、STEP 7－Micro/WIN 软件的使用、PLC 的编程语言、编程方法与调试；第二部分为应用部分，包括 PLC 的通信、PLC 在过程控制的应用、PLC 在变频调速中的应用和运动控制等。

本书内容丰富，重点突出，善于用例子诠释重点难点，共有典型例题 140 多个，便于读者模仿学习，另外每章配有习题供读者训练之用。大部分实例都有详细的软件、硬件配置清单，并配有接线图和程序。

本书可以作为学习 S7－200 系列 PLC 的工程技术人员的参考书，也可以作为大中专院校的机电类、信息类专业的教材。

图书在版编目（CIP）数据

S7-200 PLC 基础及工程应用／向晓汉主编．—北京：机械工业出版社，2014.7

（电气信息工程丛书）

ISBN 978-7-111-47553-8

Ⅰ．①S…　Ⅱ．①向…　Ⅲ．①可编程序控制器　Ⅳ．①TM571.6

中国版本图书馆 CIP 数据核字（2014）第 170065 号

机械工业出版社（北京市百万庄大街 22 号　邮政编码 100037）
策划编辑：时　静
责任编辑：时　静　　刘　悦
责任校对：张艳霞
责任印制：李　洋
三河市宏达印刷有限公司印刷

2014 年 9 月第 1 版·第 1 次
184mm×260mm·28.75 印张·711 千字
0001－3500 册
标准书号：ISBN 978-7-111-47553-8
定价：69.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务
社服务中心：（010）88361066
销售一部：（010）68326294
销售二部：（010）88379649
读者购书热线：（010）88379203

网络服务
教材网：http://www.cmpedu.com
机工官网：http://www.cmpbook.com
机工官博：http://weibo.com/cmp1952
封面无防伪标均为盗版

前　　言

随着计算机技术的发展，以可编程序控制器、变频器调速和计算机通信等技术为主体的新型电气控制系统已经逐渐取代传统的继电器电气控制系统，并广泛应用于各行业。由于西门子 S7－200 系列 PLC 具有很高的性价比，因此在工控市场占有非常大的份额，应用十分广泛。本书力求尽可能简单和详细，基础部分用较多的小例子引领读者入门，让读者学完后，能完成简单的工程；应用部分精选工程的实际案例，供读者模仿学习，提高读者解决实际问题的能力。为了使读者能更好地掌握相关知识，我们在总结长期的教学经验和工程实践的基础上，联合相关企业人员，共同编写了本书，力争使读者通过“看书”就能学会 S7－200 系列 PLC。

在编写过程中，我们将一些生动的操作实例融入到教材中，以提高读者的学习兴趣。本书与其他相关书籍相比，具有以下特点：

1）用实例引导读者学习。该书的大部分章节用精选的例子讲解，全书共有实例 140 多个。例如，用例子说明通信的实现的全过程。

2）重点的例子都包含软硬件的配置方案图、接线图和程序。为确保程序的正确性，程序已经在 PLC 上运行通过。

3）本书实用，实例易于工程移植。

全书共分 11 章。第 4、5、6 章由无锡职业技术学院的向晓汉编写；第 1 章由唐克彬编写；第 2 章由无锡雷华科技有限公司的欧阳慧编写；第 3、8 章由无锡雪浪环保科技有限公司的王飞飞编写；第 7 章由无锡雷华科技有限公司的陆彬编写；第 9 章由无锡雪浪环保科技有限公司的刘摇摇编写；第 10、11 章由桂林电子科技大学的向定汉编写。参与编写的还有李润海、苏高峰和曹英强等。本书由向晓汉任主编，陆金荣高级工程师任主审。

由于编者水平有限，缺点和错误在所难免，敬请读者批评指正，编者将万分感激！

作　者

2014 年 3 月

目　　录

前言
第 1 章　可编程序控制器（PLC）基础 …… 1
1.1　概述 …… 1
1.1.1　PLC 的发展历史 …… 1
1.1.2　PLC 的主要特点 …… 2
1.1.3　PLC 的应用范围 …… 3
1.1.4　PLC 的分类与性能指标 …… 3
1.1.5　PLC 与继电器控制系统的比较 …… 4
1.1.6　PLC 与微机的比较 …… 5
1.1.7　PLC 的发展趋势 …… 5
1.1.8　PLC 在我国 …… 6
1.2　可编程序控制器的结构和工作原理 …… 6
1.2.1　可编程序控制器的硬件组成 …… 6
1.2.2　可编程序控制器的工作原理 …… 11
1.2.3　可编程序控制器的立即输入、输出功能 …… 12
1.3　接近开关 …… 13
1.3.1　接近开关的功能 …… 13
1.3.2　接近开关的分类和工作原理 …… 13
1.3.3　接近开关的选型 …… 14
1.3.4　应用接近开关的注意事项 …… 16
小结 …… 19
习题 …… 19
第 2 章　S7－200 系列 PLC …… 20
2.1　S7－200 系列 PLC 简介 …… 20
2.1.1　西门子 S7 系列模块简介 …… 20
2.1.2　S7－200 PLC 的性能特点 …… 21
2.2　S7－200 CPU 模块及其接线 …… 22
2.2.1　S7－200 CPU 模块 …… 22
2.2.2　S7－200 CPU 的接线 …… 23
2.3　S7－200 PLC 扩展模块 …… 26
2.3.1　数字量 I/O 扩展模块 …… 27
2.3.2　模拟量 I/O 扩展模块 …… 28
2.3.3　其他扩展模块 …… 32
2.4　S7－200 PLC 的安装 …… 34
2.4.1　安装的预留空间 …… 34

2.4.2 S7－200 PLC 模块的安装和拆卸 …… 35
2.5 电源需求 …… 36
2.5.1 最大 I/O 配置 …… 36
2.5.2 电源需求计算 …… 37
小结 …… 37
习题 …… 37
第 3 章 西门子 S7－200 PLC 编程软件 …… 39
3.1 STEP 7－Micro/WIN 编程软件的安装 …… 39
3.1.1 STEP 7－Micro/WIN 编程软件概述 …… 39
3.1.2 STEP 7－Micro/WIN 编程软件的安装步骤 …… 40
3.2 STEP 7－Micro/WIN 的使用 …… 43
3.2.1 STEP 7－Micro/WIN 软件的打开 …… 43
3.2.2 STEP 7－Micro/WIN 软件的界面介绍 …… 43
3.2.3 创建新工程 …… 47
3.2.4 保存工程 …… 48
3.2.5 打开工程 …… 48
3.2.6 系统块的设置 …… 49
3.2.7 数据块 …… 52
3.2.8 程序调试 …… 53
3.2.9 交叉引用 …… 55
3.2.10 符号表 …… 55
3.2.11 变量表 …… 56
3.2.12 工具浏览条 …… 57
3.2.13 帮助菜单 …… 57
3.3 用户自定义指令库 …… 58
3.4 S7－200 PLC 扩展模块的地址分配 …… 60
3.4.1 模块的地址分配 …… 60
3.4.2 模块的地址查询 …… 61
3.5 用 STEP 7－Micro/WIN 建立一个完整的项目 …… 62
3.6 仿真软件的使用 …… 69
3.6.1 仿真软件简介 …… 69
3.6.2 仿真软件 S7－200 SIM 2.0 的使用 …… 69
小结 …… 70
习题 …… 70
第 4 章 PLC 的编程语言 …… 72
4.1 S7－200 PLC 的编程基础知识 …… 72
4.1.1 数据的存储类型 …… 72
4.1.2 元件的功能与地址分配 …… 74
4.1.3 STEP 7 中的编程语言 …… 79
4.2 位逻辑指令 …… 80
4.2.1 基本位操作指令 …… 80

4.2.2 置位/复位指令 …… 83
4.2.3 RS 触发器指令 …… 83
4.2.4 边沿触发指令 …… 85
4.2.5 逻辑栈操作指令 …… 87
4.3 定时器与计数器指令 …… 88
4.3.1 定时器指令 …… 88
4.3.2 计数器指令 …… 96
4.3.3 基本指令的应用实例 …… 102
4.4 功能指令 …… 114
4.4.1 比较指令 …… 114
4.4.2 数据处理指令 …… 117
4.4.3 移位与循环指令 …… 121
4.4.4 时钟指令及其应用 …… 126
4.4.5 算术运算指令 …… 129
4.4.6 功能指令的应用 …… 142
4.5 S7-200 PLC 的程序控制指令及其应用 …… 145
4.5.1 子程序调用 …… 145
4.5.2 中断调用 …… 147
4.5.3 跳转指令 …… 152
4.5.4 循环指令 …… 152
4.5.5 暂停指令 …… 153
4.5.6 结束指令 …… 154
4.5.7 指针 …… 154
4.5.8 顺控继电器指令 …… 155
4.5.9 程序控制指令的应用 …… 157
小结 …… 157
习题 …… 157
第 5 章 逻辑控制编程的编写方法 …… 161
5.1 顺序功能图 …… 161
5.1.1 顺序功能图的画法 …… 161
5.1.2 梯形图编程的原则 …… 166
5.2 逻辑控制的梯形图编程方法 …… 168
小结 …… 187
习题 …… 187
第 6 章 S7-200 PLC 的通信及其应用 …… 189
6.1 通信基础知识 …… 189
6.1.1 通信的基本概念 …… 189
6.1.2 RS-485 标准串行接口 …… 190
6.1.3 PLC 网络的术语解释 …… 191
6.1.4 OSI 参考模型 …… 192
6.1.5 工业通信网络结构 …… 193

6.1.6 西门子通信网络技术说明 …… 194
6.2 PPI 通信及其应用 …… 195
6.2.1 PPI 通信基础 …… 195
6.2.2 PPI 通信的应用 …… 196
6.3 S7-200 PLC 的 MPI 通信及应用 …… 204
6.3.1 MPI 通信概述 …… 204
6.3.2 无组态连接 MPI 通信的应用 …… 204
6.4 自由口通信 …… 211
6.4.1 自由口通信概述 …… 211
6.4.2 S7-200 系列 PLC 自由口通信的应用 …… 213
6.4.3 S7-200 PLC 与个人计算机（自编程序）的自由口通信 …… 218
6.4.4 S7-200 PLC 与三菱 FX 系列 PLC 的自由口通信 …… 220
6.5 PROFIBUS 现场总线及应用 …… 224
6.5.1 现场总线的概念 …… 224
6.5.2 主流现场总线的简介 …… 224
6.5.3 现场总线的特点 …… 226
6.5.4 现场总线的现状 …… 227
6.5.5 现场总线的发展 …… 227
6.5.6 S7-200 PLC PROFIBUS 通信的应用 …… 228
6.6 以太网通信 …… 239
6.6.1 工业以太网通信简介 …… 239
6.6.2 S7-200 PLC 以太网通信的应用 …… 240
6.7 Modbus 通信 …… 253
6.7.1 Modbus 通信概述 …… 253
6.7.2 S7-200 PLC Modbus 通信的应用 …… 255
小结 …… 259
习题 …… 260
第 7 章 上位机对 S7-200 系列 PLC 的监控 …… 261
7.1 简单组态软件工程的建立 …… 261
7.1.1 认识组态软件 …… 261
7.1.2 安装 WinCC 的硬件要求 …… 262
7.1.3 安装 WinCC 的软件要求 …… 263
7.2 用 WinCC 监控 S7-200 PLC …… 264
7.2.1 OPC 基本知识 …… 264
7.2.2 PC Access 软件简介 …… 265
7.2.3 WinCC 与 S7-200 PLC 的通信 …… 267
小结 …… 275
习题 …… 275
第 8 章 S7-200 PLC 的高速输入和高速输出功能及其应用 …… 276
8.1 S7-200 PLC 的高速输入及其应用 …… 276
8.1.1 高速计数器的简介 …… 276

8.1.2 高速计数器在转速测量中的应用 …… 281
8.2 S7-200 PLC 的高速输出及其应用 …… 283
8.2.1 S7-200 PLC 的高速输出简介 …… 283
8.2.2 PWM 的应用 …… 285
8.2.3 S7-200 系列 PLC 的高速输出点控制步进电动机 …… 286
8.2.4 使用定位模块控制步进电动机 …… 296
8.2.5 步进电动机的调速控制 …… 307
8.2.6 步进电动机的正反转控制 …… 309
8.2.7 PLC 控制伺服系统 …… 311
小结 …… 315
习题 …… 315
第 9 章 PLC 在变频器调速系统中的应用 …… 317
9.1 三相异步电动机的机械特性和调速原理 …… 317
9.1.1 三相异步电动机的机械特性 …… 317
9.1.2 三相异步电动机的调速原理 …… 318
9.1.3 变频器的发展 …… 321
9.1.4 变频器的分类 …… 324
9.2 西门子 MM 440 变频器使用简介 …… 326
9.2.1 认识变频器 …… 326
9.2.2 西门子 MM 440 变频器使用简介 …… 327
9.3 变频器多段速度给定 …… 332
9.4 变频器模拟量速度给定 …… 339
9.4.1 模拟量模块的简介 …… 339
9.4.2 模拟量速度给定的应用 …… 340
9.5 变频器的通信速度给定 …… 343
9.5.1 USS 协议简介 …… 343
9.5.2 USS 通信的应用 …… 344
9.6 使用变频器时的常用回路 …… 349
9.6.1 制动控制回路 …… 349
9.6.2 使用变频器时，电动机的起停控制 …… 350
9.6.3 变频器的正反转控制 …… 351
小结 …… 353
习题 …… 353
第 10 章 S7-200 系列 PLC 的其他应用技术 …… 354
10.1 S7-200 PLC 在 PID 中的应用 …… 354
10.1.1 PID 控制原理简介 …… 354
10.1.2 利用 S7-200 PLC 进行电炉的温度控制 …… 357
10.2 程序的下载方法 …… 366
10.2.1 用 PPI 协议下载程序 …… 366
10.2.2 用 MPI 协议下载程序 …… 373
10.2.3 用 PROFIBUS 协议下载程序 …… 376

10.2.4 用 TCP/IP 下载程序 …… 378
小结 …… 386
习题 …… 386
第 11 章 工程应用案例 …… 387
11.1 分高峰和非高峰时段的交通灯 PLC 控制系统 …… 387
11.1.1 绘制时序图 …… 387
11.1.2 系统软硬件配置 …… 388
11.1.3 编写控制程序 …… 388
11.2 行车呼叫 PLC 控制系统 …… 394
11.2.1 系统软硬件配置 …… 394
11.2.2 编写程序 …… 395
11.3 送料小车自动往复运动的 PLC 控制系统 …… 399
11.3.1 系统软硬件配置 …… 400
11.3.2 编写控制程序 …… 401
11.4 搬运站 PLC 控制系统 …… 405
11.4.1 系统软硬件配置 …… 406
11.4.2 编写控制程序 …… 407
11.5 跳动度测试仪控制 …… 410
11.5.1 系统软硬件配置 …… 410
11.5.2 编写控制程序 …… 412
11.6 刨床 PLC 控制系统 …… 414
11.6.1 系统软硬件配置 …… 414
11.6.2 编写控制程序 …… 415
11.7 电镀生产线的 PLC 控制 …… 417
11.7.1 系统软硬件配置 …… 418
11.7.2 编写控制程序 …… 420
11.8 十字滑台的 PLC 控制 …… 425
11.8.1 系统软硬件配置 …… 426
11.8.2 编写控制程序 …… 427
11.9 定长前切机的 PLC 控制 …… 431
11.9.1 系统软硬件配置 …… 432
11.9.2 控制程序的编写 …… 433
11.10 工业氮气管道流量监控系统的 PLC 控制 …… 440
11.10.1 系统软硬件配置 …… 440
11.10.2 编写控制程序 …… 442
11.11 物料搅拌机的 PLC 控制 …… 444
11.11.1 系统软硬件配置 …… 444
11.11.2 编写控制程序 …… 446
小结 …… 447
习题 …… 448
参考文献 …… 449

第 1 章　可编程序控制器（PLC）基础

本章介绍可编程序控制器的历史、功能、特点、应用范围、发展趋势、在我国的使用情况、结构和工作原理等知识，使读者初步了解可编程序控制器，这是学习本书后续内容的必要准备。

1.1　概述

可编程序控制器（Programmable Logic Controller）简称 PLC，国际电工委员会（IEC）于 1985 年对可编程序控制器作了如下定义："可编程序控制器是一种数字运算操作的电子系统，专为在工业环境下应用而设计。它采用可编程序的存储器，用来在其内部存储执行逻辑运算、顺序控制、定时、计数和算术运算等操作的指令，并通过数字、模拟的输入和输出，控制各种类型的机械或生产过程。可编程序控制器及其有关设备，都应按易于与工业控制系统连成一个整体，易于扩充功能的原则设计。" PLC 是一种工业计算机，其种类繁多，不同厂家的产品有各自的特点，但作为工业标准设备，可编程序控制器又有一定的共性。

1.1.1　PLC 的发展历史

20 世纪 60 年代以前，汽车生产线的自动控制系统基本上都是由继电器控制装置构成的。当时每次改型都直接导致继电器控制装置的重新设计和安装，福特汽车公司的老板曾经说，无论顾客需要什么样的汽车，福特的汽车永远是黑色的，从侧面反映汽车改型和升级换代比较困难。为了改变这一现状，1969 年，美国的通用汽车公司（GM）公开招标，要求用新的装置取代继电器控制装置，并提出十项招标指标，要求编程方便、现场可修改程序、维修方便、采用模块化设计、体积小、可与计算机通信等。同一年，美国数字设备公司（DEC）研制出了世界上第一台可编程序控制器 PDP－14，在美国通用汽车公司的生产线上试用成功，并取得了满意的效果，可编程序控制器从此诞生。由于当时的 PLC 只能取代继电器接触器控制，功能仅限于逻辑运算、计时、计数等，所以称为"可编程逻辑控制器"。伴随着微电子技术、控制技术与信息技术的不断发展，可编程序控制器的功能不断增强。美国电气制造商协会（NEMA）于 1980 年正式将其命名为"可编程序控制器"，简称 PC，由于这个名称和个人计算机的简称相同，容易混淆，因此在我国，很多人仍然习惯称可编程序控制器为 PLC。

由于 PLC 具有易学易用、操作方便、可靠性高、体积小、通用灵活和使用寿命长等一系列优点，因此，很快就在工业中得到了广泛的应用。同时，这一新技术也受到其他国家的重视。1971 年日本引进这项技术，很快研制出日本第一台 PLC，欧洲于 1973 年研制出第一台 PLC，我国从 1974 年开始研制，1977 年国产 PLC 正式投入工业应用。

进入20世纪80年代以来，随着电子技术的迅猛发展，以16位和32位微处理器构成的微机化PLC得到快速发展（例如GE的RX7i，使用的是赛扬CPU，其主频达1 GHz，其信息处理能力几乎和个人计算机相当），使得PLC在设计、性能价格比以及应用方面有了突破，不仅控制功能增强，功耗和体积减小，成本下降，可靠性提高，编程和故障检测更为灵活方便，而且随着远程I/O和通信网络、数据处理和图像显示的发展，已经使得PLC普遍用于控制复杂生产过程。PLC已经成为工厂自动化的三大支柱之一。

1.1.2 PLC的主要特点

PLC之所以高速发展，除了工业自动化的客观需要外，还有许多适合工业控制的独特优点，它较好地解决了工业控制领域中普遍关心的可靠、安全、灵活、方便、经济等问题，其主要特点如下：

1. 抗干扰能力强，可靠性高

在传统的继电器控制系统中，使用了大量的中间继电器、时间继电器，由于器件的固有缺点，如器件老化、接触不良、触点抖动等现象，大大降低了系统的可靠性。而在PLC控制系统中大量的开关动作由无触点的半导体电路完成，因此故障大大减少。

此外，PLC的硬件和软件方面采取了措施，提高其可靠性。在硬件方面，所有的I/O接口都采用了光电隔离，使得外部电路与PLC内部电路实现了物理隔离。各模块都采用了屏蔽措施，以防止辐射干扰。电路中采用了滤波技术，以防止或抑制高频干扰。在软件方面，PLC具有良好的自诊断功能，一旦系统的软、硬件发生异常情况，CPU会立即采取有效措施，以防止故障扩大。通常PLC具有看门狗功能。

对于大型的PLC系统，还可以采用双CPU构成冗余系统或者三CPU构成表决系统，使系统的可靠性进一步提高。

2. 程序简单易学，系统的设计调试周期短

PLC是面向用户的设备，PLC的生产厂家充分考虑到现场技术人员的技能和习惯，可采用梯形图或面向工业控制的简单指令形式。梯形图与继电器控制原理图很相似，直观、易懂、易掌握，不需要学习专门的计算机知识和语言。设计人员可以在设计室设计、修改和模拟调试程序，非常方便。

3. 安装简单，维修方便

PLC不需要专门的机房，可以在各种工业环境下直接运行，使用时只需将现场的各种设备与PLC相应的I/O端相连接，即可投入运行。各种模块上均有运行和故障指示装置，便于用户了解运行情况和查找故障。

4. 采用模块化结构，体积小，重量轻

为了适应工业控制需求，除了整体式PLC外，绝大多数PLC采用模块化结构。PLC的各部件，包括CPU、电源、I/O等都采用模块化设计。此外，PLC相对于通用工控机，其体积和重量要小得多。

5. 丰富的I/O接口模块，扩展能力强

PLC针对不同的工业现场信号（如交流或直流、开关量或模拟量、电压或电流、脉冲或电位、强电或弱电等）有相应的I/O模块与工业现场的器件或设备（如按钮、行程开关、

接近开关、传感器及变送器、电磁线圈、控制阀等）直接连接。另外，为了提高操作性能，它还有多种人——机对话的接口模块，为了组成工业局部网络，它还有多种通信联网的接口模块等。

1.1.3 PLC 的应用范围

目前，PLC 在国内外已广泛应用于专用机床、控制系统、自动化楼宇、钢铁、石油、化工、电力、建材、汽车、纺织机械、交通运输、环保以及文化娱乐等各行各业。随着 PLC 性能价格比的不断提高，其应用范围还将不断扩大，其应用场合可以说是无处不在，具体应用大致可归纳为如下几类：

1）顺序控制。这是 PLC 最基本、最广泛应用的领域，它取代传统的继电器顺序控制，PLC 用于单机控制、多机群控制、自动化生产线的控制。例如数控机床、注塑机、印刷机械、电梯控制和纺织机械等。

2）计数和定时控制。PLC 为用户提供了足够的定时器和计数器，并设置相关的定时和计数指令，PLC 的计数器和定时器精度高、使用方便，可以取代继电器控制系统中的时间继电器和计数器。

3）位置控制。大多数的 PLC 制造商，目前都提供拖动步进电动机或伺服电动机的单轴或多轴位置控制模块，这一功能可广泛用于各种机械，如金属切削机床、装配机械等。

4）模拟量处理。PLC 通过模拟量的输入/输出模块，实现模拟量与数字量的转换，并对模拟量进行控制，有的还具有 PID 控制功能。例如用于锅炉的水位、压力和温度控制。

5）数据处理。现代的 PLC 具有数学运算、数据传递、转换、排序和查表等功能，也能完成数据的采集、分析和处理。

6）通信联网。PLC 的通信包括 PLC 相互之间、PLC 与上位计算机、PLC 和其他智能设备之间的通信。PLC 系统与通用计算机可以直接或通过通信处理单元、通信转接器相连构成网络，以实现信息的交换，并可构成“集中管理、分散控制”的分布式控制系统，满足工厂自动化系统的需要。

1.1.4 PLC 的分类与性能指标

1. PLC 的分类

（1）从组成结构形式分类

可以将 PLC 分为两类：一类是整体式 PLC（也称单元式），其特点是电源、中央处理单元、I/O 接口都集成在一个机壳内；另一类是标准模板式结构化的 PLC（也称组合式），其特点是电源模板、中央处理单元模板、I/O 模板等在结构上是相互独立的，可根据具体的应用要求，选择合适的模块，安装在固定的机架或导轨上，构成一个完整的 PLC 应用系统。

（2）按 I/O 点容量分类

① 小型 PLC。小型 PLC 的 I/O 点数一般在 128 点以下。较为常见的有三菱的 FX 系列 PLC 和西门子的 S7－200 系列 PLC 等。

② 中型 PLC。中型 PLC 采用模块化结构，其 I/O 点数一般在 256～1 024 点之间。较为

常见的有西门子的 S7－300 系列 PLC 和 GE 的 RX3i 系列等。

③ 大型 PLC。一般 I/O 点数在 1 024 点以上的称为大型 PLC。较为常见的有西门子的 S7－400 系列 PLC 和 GE 的 RX7i 系列等。

2. PLC 的性能指标

各厂家的 PLC 虽然各有特色，但其主要性能指标是相同的。

（1）输入/输出（I/O）点数

输入/输出（I/O）点数是最重要的一项技术指标，是指 PLC 的面板上连接外部输入、输出端子数，常称为“点数”，用输入与输出点数的和表示。点数越多表示 PLC 可接入的输入器件和输出器件越多，控制规模越大。点数是 PLC 选型时最重要的指标之一。

（2）扫描速度

扫描速度是指 PLC 执行程序的速度。以 ms/K 步为单位，即执行 1K 步指令所需的时间。l 步占 1 个地址单元。

（3）存储容量

存储容量通常用 K 字（KW）或 K 字节（KB）、K 位来表示。这里 1 K＝1 024。有的 PLC 用“步”来衡量，一步占用一个地址单元。存储容量表示 PLC 能存放多少用户程序。例如，三菱型号为 FX2N－48MR 的 PLC 存储容量为 8 000 步。有的 PLC 的存储容量可以根据需要配置，有的 PLC 的存储器可以扩展。

（4）指令系统

指令系统表示该 PLC 软件功能的强弱。指令越多，编程功能就越强。

（5）内部寄存器（继电器）

PLC 内部有许多寄存器用来存放变量、中间结果、数据等，还有许多辅助寄存器可供用户使用。因此寄存器的配置也是衡量 PLC 功能的一项指标。

（6）扩展能力

扩展能力是反映 PLC 性能的重要指标之一。PLC 除了主控模块外，还可配置实现各种特殊功能的高功能模块。例如 A－D 模块、D－A 模块、高速计数模块、远程通信模块等。

1.1.5 PLC 与继电器控制系统的比较

在 PLC 出现以前，继电器硬接线电路是逻辑、顺序控制的唯一执行者，它结构简单、价格低廉，一直被广泛应用。PLC 出现后，几乎所有的方面都超过继电器控制系统，两者的性能比较见表 1-1。

表 1-1 可编程序控制器与继电器控制系统性能的比较

序号	比较项目	继电器控制	可编程序控制器控制
1	控制逻辑	硬接线多、体积大、连线多	软逻辑、体积小、接线少、控制灵活
2	控制速度	通过触点开关实现控制，动作受继电器硬件限制，通常超过 10 ms	由半导体电路实现控制，指令执行时间段，一般为微秒级
3	定时控制	由时间继电器控制，精度差	由集成电路的定时器完成，精度高
4	设计与施工	设计、施工、调试必须按照顺序进行，周期长	系统设计完成后，施工与程序设计同时进行，周期短

（续）

序号	比 较 项 目	继电器控制	可编程序控制器控制
5	可靠性与维护	继电器的触点寿命短，可靠性和维护性差	无触点，寿命长，可靠性高，有自诊断功能
6	价格	价格低	价格高

1.1.6　PLC 与微机的比较

采用微电子技术制造的可编程序控制器与微机一样，也由 CPU、ROM（或者 FLASH）、RAM、I/O 接口等组成，但又不同于一般的微机，可编程序控制器采用了特殊的抗干扰技术，是一种特殊的工业控制计算机，更加适合工业控制。两者的性能比较见表 1-2。

表 1-2　PLC 与微机的比较

序号	比 较 项 目	可编程序控制器控制	微机控制
1	应用范围	工业控制	科学计算、数据处理、计算机通信
2	使用环境	工业现场	具有一定温度和湿度的机房
3	输入/输出	控制强电设备，需要隔离	与主机弱电联系，不隔离
4	程序设计	一般使用梯形图语言，易学易用	编程语言丰富，如 C、BASIC 等
5	系统功能	自诊断、监控	使用操作系统
6	工作方式	循环扫描方式和中断方式	中断方式

1.1.7　PLC 的发展趋势

PLC 发展趋势主要有以下几个方面：

1）向高性能、高速度、大容量发展。

2）网络化。强化通信能力和网络化，向下将多个可编程序控制器或者多个 I/O 框架相连；向上与工业计算机、以太网等相连，构成整个工厂的自动化控制系统。即便是微型的 S7-200 系列 PLC 也能组成多种网络，通信功能十分强大。

3）小型化、低成本、简单易用。目前，有的小型 PLC 的价格只有几百元人民币。

4）不断提高编程软件的功能。编程软件可以对 PLC 控制系统的硬件组态，在屏幕上可以直接生成和编辑梯形图、指令表、功能块图和顺序功能图程序，并可以实现不同编程语言的相互转换。程序可以下载、存盘和打印，通过网络或电话线，还可以实现远程编程。

5）适合 PLC 应用的新模块。随着科技的发展，对工业控制领域将提出更高的、更特殊的要求，因此，必须开发特殊功能模块来满足这些要求。

6）PLC 的软件化与 PC 化。目前已有多家厂商推出了在 PC 上运行的可实现 PLC 功能的软件包，也称为“软 PLC”，“软 PLC”的性能价格比比传统的“硬 PLC”更高，是 PLC 的一个发展方向。

PC 化的 PLC 类似于 PLC，但它采用了 PC 的 CPU，功能十分强大，如 GE 的 Rx7i 和 Rx3i 使用的就是工控机用的赛扬 CPU，主频已经达到 1 GHz。

1.1.8 PLC 在我国

1. 国外 PLC 品牌

目前 PLC 在我国得到了广泛的应用，很多知名厂家的 PLC 在我国都有应用。

1）美国是 PLC 生产大国，有 100 多家 PLC 生产厂家。其中 A－B 公司的 PLC 产品规格比较齐全，主推大中型 PLC，主要产品系列是 PLC－5。通用电气公司也是知名 PLC 生产厂商，大中型 PLC 产品系列有 RX3i 和 RX7i 等。德州仪器公司也生产大、中、小全系列 PLC 产品。

2）欧洲的 PLC 产品也久负盛名。德国的西门子公司、AEG 公司和法国的 TE（施耐德）公司都是欧洲著名的 PLC 制造商。其中西门子公司的 PLC 产品与美国的 A－B 公司的 PLC 产品齐名。

3）日本的小型 PLC 具有一定的特色，性价比较高，比较知名的品牌有三菱、欧姆龙、松下、富士、日立和东芝等，在小型机市场，日系 PLC 的市场份额曾经高达 70%。

2. 国产 PLC 品牌

我国自主品牌的 PLC 生产厂家有近 30 余家。在目前已经上市的众多 PLC 产品中，还没有形成规模化的生产和名牌产品。单从技术角度来看，国产小型 PLC 与国际知名品牌小型 PLC 差距正在缩小，使用越来越多。例如和利时、深圳汇川和无锡信捷等公司生产的微型 PLC 已经比较成熟，其可靠性在许多低端应用中得到了验证，逐渐被用户认可，但其知名度与世界先进水平还有一定的差距。

总的来说，我国使用的小型可编程序控制器主要以日本和国产的品牌为主，而大中型可编程序控制器主要以欧美的品牌为主。目前 95% 以上的 PLC 市场被国外品牌所占领。

1.2 可编程序控制器的结构和工作原理

1.2.1 可编程序控制器的硬件组成

可编程序控制器种类繁多，但其基本结构和工作原理相同。可编程序控制器的功能结构由 CPU（中央处理器）、存储器和输入模块/输出模块三部分组成，如图 1-1 所示。

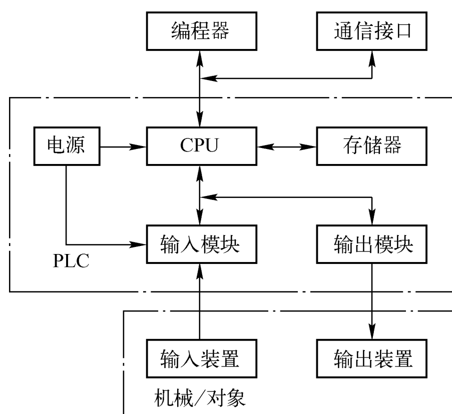


图 1-1 可编程序控制器结构框图

1. CPU（中央处理器）

CPU 的功能是完成 PLC 内所有的控制和监视操作。中央处理器一般由控制器、运算器和寄存器组成。CPU 通过数据总线、地址总线和控制总线与存储器、输入输出接口电路连接。

2. 存储器

在 PLC 中使用两种类型的存储器：一种是只读类型的存储器，如 EPROM 和 EEPROM；另一种是可读/写的随机存储器 RAM。PLC 的存储器分为 5 个区域，如图 1-2 所示。

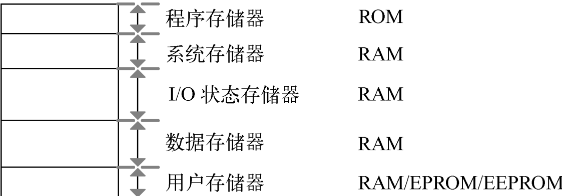


图 1-2　存储器的区域划分

程序存储器的类型是只读存储器（ROM），PLC 的操作系统存放在这里，程序由制造商固化，通常不能修改。有的 PLC 厂商对部分 PLC 产品（如西门子的 S7－1200 和 S7－200 SMART）提供操作系统升级服务。存储器中的程序负责解释和编译用户编写的程序、监控 I/O 口的状态、对 PLC 进行自诊断、扫描 PLC 中的程序等。系统存储器属于随机存储器（RAM），主要用于存储中间计算结果和数据、系统管理，有的 PLC 厂家用系统存储器存储一些系统信息，如错误代码等。系统存储器不对用户开放。I/O 状态存储器属于随机存储器，用于存储 I/O 装置的状态信息，每个输入模块和输出模块都在 I/O 映像表中分配一个地址，而且这个地址是唯一的。数据存储器属于随机存储器，主要用于数据处理功能，为计数器、定时器、算术计算和过程参数提供数据存储。有的厂家将数据存储器细分为固定数据存储器和可变数据存储器。用户存储器的类型可以是随机存储器、可擦除存储器（EPROM）和电擦除存储器（EEPROM），高档的 PLC 还可以用 FLASH。用户存储器主要用于存放用户编写的程序。存储器的关系如图 1-3 所示。

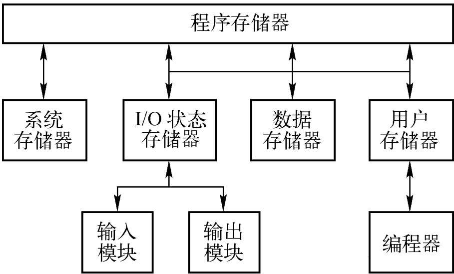


图 1-3　存储器的关系

只读存储器可以用来存放系统程序，PLC 断电后再上电，系统内容不变且重新执行。只读存储器也可用来固化用户程序和一些重要参数，以免因偶然操作失误而造成程序和数据的破坏或丢失。随机存储器中一般存放用户程序和系统参数。当 PLC 处于编程工作时，CPU 从 RAM 中取指令并执行。用户程序执行过程中产生的中间结果也在 RAM 中暂时存放。

RAM 通常由 CMOS 型集成电路组成，功耗小，但断电时内容消失，所以一般使用大电容或后备锂电池保证掉电后 PLC 的内容在一定时间内不丢失。

3. 输入/输出接口

可编程序控制器的输入和输出信号可以是开关量或模拟量。输入/输出接口是 PLC 内部弱电（low power）信号和工业现场强电（high power）信号联系的桥梁。输入/输出接口主要有两个作用：一是利用内部的电隔离电路将工业现场和 PLC 内部进行隔离，起保护作用；二是调理信号，可以把不同的信号（如强电、弱电信号）调理成 CPU 可以处理的信号（5 V、3. 3 V 或 2. 7 V 等），如图 1-4 所示。

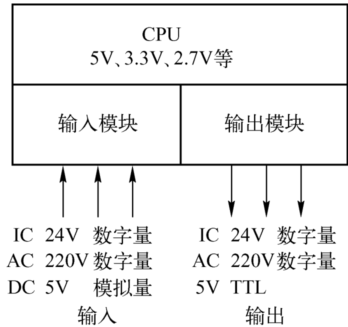


图 1-4　输入/输出接口

输入/输出接口模块是 PLC 系统中最大的部分，输入/输出接口模块通常需要电源，输入电路的电源可以由外部提供，对于模块化的 PLC 还需要背板（安装机架）。

（1）输入接口电路

1）输入接口电路的组成和作用。输入接口电路由接线端子、输入调理和电平转换电路、模块状态显示、电隔离电路和多路选择开关模块组成，如图 1-5 所示。现场的信号必须连接在输入端子才可能将信号输入到 CPU 中，它提供了外部信号输入的物理接口；调理和电平转换电路十分重要，可以将工业现场的信号（如强电 AC 220 V 信号）转化成电信号（CPU 可以识别的弱电信号）；电隔离电路主要利用电隔离器件将工业现场的机械或者电输入信号和 PLC 的 CPU 的信号隔开，它能确保过高的电干扰信号和浪涌不串入 PLC 的微处理器，起保护作用，有三种隔离方式，用得最多的是光电隔离，其次是变压器隔离和干簧继电器隔离；当外部有信号输入时，输入模块上有指示灯显示，这个电路比较简单，当线路中有故障时，它帮助用户查找故障，由于氖灯或 LED 灯的寿命比较长，所以这个灯通常是氖灯或 LED 灯；多路选择开关接受调理完成的输入信号，并存储在多路开关模块中，当输入循环扫描时，多路开关模块中信号输送到 I/O 状态寄存器中。

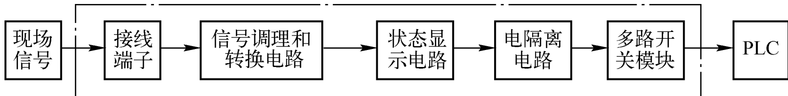


图 1-5　输入接口的结构

2）输入信号的设备的种类。输入信号可以是离散信号和模拟信号。当输入端是离散信号时，输入端的设备类型可以是限位开关、按钮、压力继电器、继电器触点、接近开关、选择开关、光电开关等，如图 1-6 所示。当输入为模拟量输入时，输入设备的类型可以是压

力传感器、温度传感器、流量传感器、电压传感器、电流传感器、力传感器等。输入接口的内部电路如图 1-6 所示。

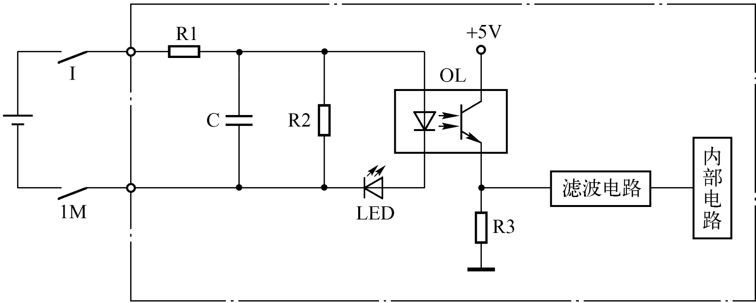


图 1-6　输入接口的内部电路

【关键点】从图 1-6 可以看到：数字量输入回路有光电隔离，所以一般数字量输入回路只要不把较大的电压（如 AC 110 V）误接入电路，错误的接线通常不会造成输入回路的烧毁。但模拟量输入回路一般不设计光电隔离，所以在接线要特别注意，不能短路，否则容易烧毁通道。

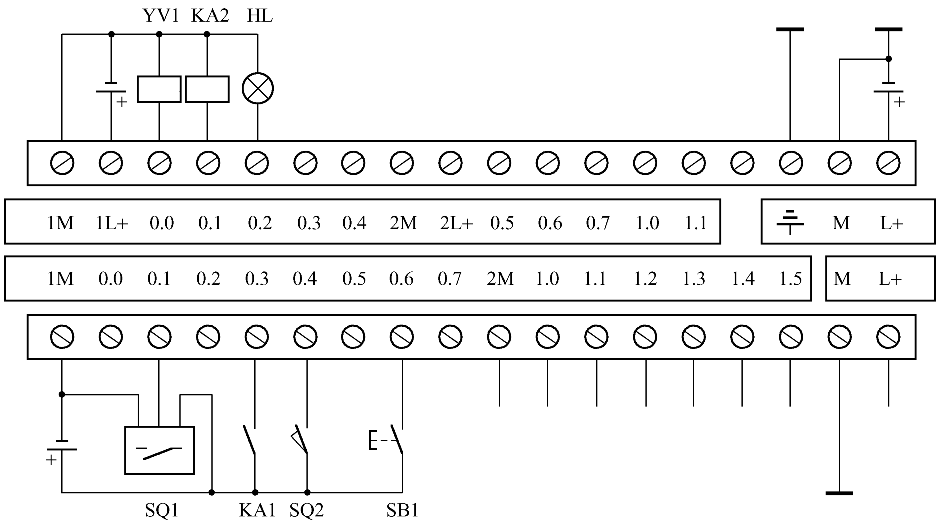


图 1-7　输入/输出接口

（2）输出接口电路

1）输出接口电路的组成和作用。输出接口电路由多路选择开关模块、信号锁存器、电隔离电路、模块状态显示、输出电平转换电路和接线端子组成，如图 1-8 所示。在输出扫描期间，多路选择开关模块接受来自映像表中的输出信号，并对这个信号的状态和目标地址进行译码，最后将信息送给锁存器；信号锁存器是将多路选择开关模块的信号保存起来，直到下一次更新；输出接口的电隔离电路作用和输入模块的一样，但是由于输出模块输出的信号比输入信号要强得多，因此要求隔离电磁干扰和浪涌的能力更高；输出电平转换电路将隔离电路送来的信号放大成足够驱动现场设备的信号，放大器件可以是双向晶闸管、晶体管和干簧继电器等；输出端的接线端子用于将输出模块与现场设备相连接。

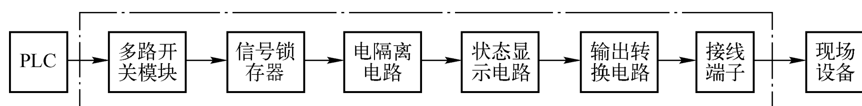


图 1-8　输出接口的结构

可编程序控制器有三种输出接口形式：继电器输出、晶体管输出和晶闸管输出形式。继电器输出形式的 PLC 的负载电源可以是直流电源或交流电源，但其输出响应频率较慢，其内部电路如图 1-9 所示。晶体管输出的 PLC 负载电源是直流电源，其输出响应频率较快，其内部电路如图 1-10 所示。晶闸管输出形式的 PLC 的负载电源是交流电源，西门子 S7-200 系列 PLC 的 CPU 模块暂时还没有晶闸管输出形式的产品出售，但三菱 FX 系列 PLC 有这种产品。选型时要特别注意 PLC 的输出形式。

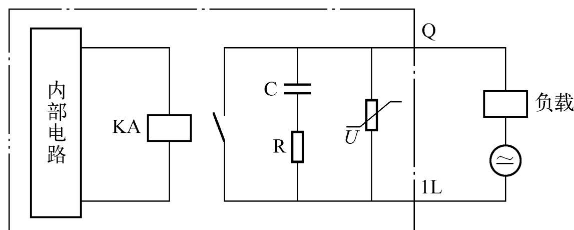


图 1-9　继电器输出内部电路

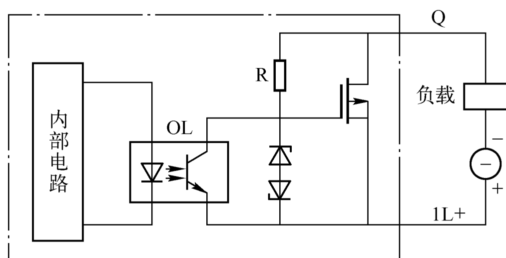


图 1-10　晶体管输出内部电路

2）输出信号的设备的种类。输出信号可以是离散信号和模拟信号。当输出端是离散信号时，输出端的设备类型可以是电磁阀的线圈、电动机起动器、控制柜的指示器、接触器线圈、LED 灯、指示灯、继电器线圈、报警器和蜂鸣器等，如图 1-7 所示。当输出为模拟量输出时，输出设备的类型可以是流量阀、AC 驱动器（如交流伺服驱动器）、DC 驱动器、模拟量仪表、温度控制器和流量控制器等。

【关键点】 PLC 的继电器型输出虽然响应速度慢，但其驱动能力强，一般为 2 A，这是继电器型输出 PLC 的一个重要的优点。有的特殊型号的 PLC，如西门子 LOGO！的某些型号驱动能力可达 5A 和 10A，能直接驱动接触器。此外，从图 1-9 可以看出继电器型输出形式的 PLC，对于一般的误接线，通常不会引起 PLC 内部器件的烧毁（高于 AC 220 V 电压是不允许的）。因此，继电器输出形式是选型时的首选，在工程实践中，用得比较多。

晶体管输出的 PLC 的输出电流一般小于 1 A，西门子 S7-200 PLC 的输出电流源是 0.75 A（西门子有的型号的 PLC 的输出电流甚至只有 0.5 A），可见晶体管输出的驱动能力较小。此外，图 1-10 可以看出晶体管型输出形式的 PLC，对于一般的误接线，可能会引起 PLC 内部器件的烧毁，所以要特别注意。

【例 1-1】某学生按如图 1-11 所示接线，之后发现按下 SB1、SB2 和 SB3 按钮，输入端的指示灯没有显示，PLC 中没有程序，但灯 HL 常亮，接线没有错误，+24 V 电源也正常。分析结果是输入和输出接口烧毁，请问分析是否正确。

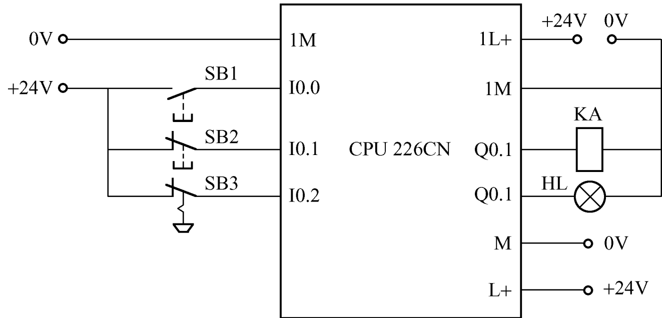


图 1-11　晶体管输出内部电路

【解】

分析如下：

① 一般输入端口不会烧毁，因为输入接口电路有光电隔离电路保护，除非有较大电压（如 AC 220 V）的误接入，而且烧毁输入接口一般也不会所有的接口同时烧毁。经过检查，发现接线端子 1 M 是“虚接”，压紧此接线端子后，输入端恢复正常。

② 误接线容易造成晶体管输出回路的器件烧毁，晶体管的击穿会造成回路导通，从而造成 HL 灯常亮。

【关键点】本书中所有的 PNP 输入和 NPN 输入都是以传感器为对象，有的资料以 PLC 为对象则变成 NPN 输入和 PNP 输入，请读者注意。

1.2.2　可编程序控制器的工作原理

PLC 是一种存储程序的控制器。用户根据某一对象的具体控制要求，编制好控制程序后，用编程器将程序输入到 PLC（或用计算机下载到 PLC）的用户程序存储器中寄存。PLC 的控制功能就是通过运行用户程序来实现的。

PLC 运行程序的方式与微型计算机相比有较大的不同，微型计算机运行程序时，一旦执行到 END 指令，程序运行结束。而 PLC 从 0 号存储地址所存放的第一条用户程序开始，在无中断或跳转的情况下，按存储地址号递增的方向顺序逐条执行用户程序，直到 END 指令结束。然后再从头开始执行，并周而复始地重复，直到停机或从运行（RUN）切换到停止（STOP）工作状态。把 PLC 这种执行程序的方式称为扫描工作方式。每扫描完一次程序就构成一个扫描周期。另外，PLC 对输入、输出信号的处理与微型计算机不同。微型计算机对输入、输出信号实时处理，而 PLC 对输入、输出信号是集中批处理。下面具体介绍 PLC 的扫描工作过程。其运行和信号处理示意如图 1-12 所示。

PLC 扫描工作方式主要分为三个阶段：输入扫描、程序执行、输出刷新。

1. 输入扫描

PLC 在开始执行程序之前，首先扫描输入端子，按顺序将所有输入信号，读入到寄存器——输入状态的输入映像寄存器中，这个过程称为输入扫描。PLC 在运行程序时，所需的输入信号不是现时取输入端子上的信息，而是取输入映像寄存器中的信息。在本工作周期内这

个采样结果的内容不会改变，只有到下一个扫描周期输入扫描阶段才被刷新。PLC 的扫描速度很快，取决于 CPU 的时钟速度。

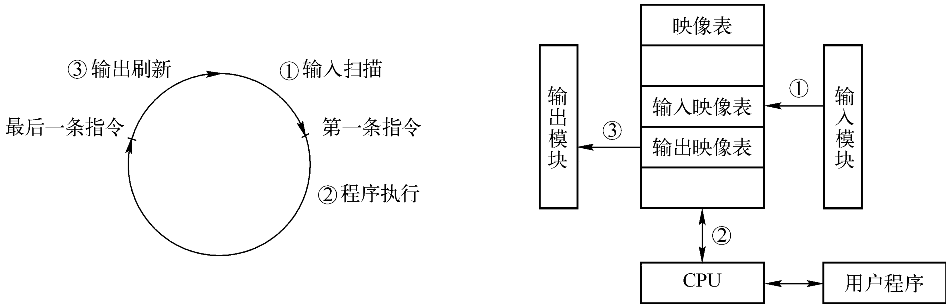


图 1-12　PLC 内部运行和信号处理示意图

2. 程序执行

PLC 完成了输入扫描工作后，按顺序从 0 号地址开始的程序进行逐条扫描执行，并分别从输入映像寄存器、输出映像寄存器以及辅助继电器中获得所需的数据进行运算处理。再将程序执行的结果写入输出映像寄存器中保存。但这个结果在全部程序未被执行完毕之前不会送到输出端子上，也就是物理输出是不会改变的。扫描时间取决于程序的长度、复杂程度和 CPU 的功能。

3. 输出刷新

在执行到 END 指令，即执行完用户所有程序后，PLC 上将输出映像寄存器中的内容送到输出锁存器中进行输出，驱动用户设备。扫描时间取决于输出模块的数量。

从以上的介绍可以知道，PLC 程序扫描特性决定了 PLC 的输入和输出状态并不能在扫描的同时改变，例如一个按钮开关的输入信号的输入刚好在输入扫描之后，那么这个信号只有在下一个扫描周期才能被读入。

上述三个步骤是 PLC 的软件处理过程，可以认为就是程序扫描时间。扫描时间通常由三个因素决定：一是 CPU 的时钟速度，越高档的 CPU，时钟速度越高，扫描时间越短；二是 I/O 模块的数量，模块数量越少，扫描时间越短；三是程序的长度，程序长度越短，扫描时间越短。一般的 PLC 执行容量为 1 K 步的程序需要的扫描时间是 1 ~ 10 ms。

1.2.3　可编程序控制器的立即输入、输出功能

比较高档的 PLC 都有立即输入、输出功能。

1. 立即输出功能

所谓立即输出功能就是输出模块在处理用户程序时，能立即被刷新。PLC 临时挂起（中断）正常运行的程序，将输出映像表中的信息输送到输出模块，立即进行输出刷新，然后再回到程序中继续运行，立即输出的示意图如图 1-13 所示。注意，立即输出功能并不能立即刷新所有的输出模块。

2. 立即输入功能

立即输入适用于要求对反应速度很严格的场合，例如几毫秒的时间对于控制来说十分关键的情况下。立即输入时，PLC 立即挂起正在执行的程序，扫描输入模块，然后更新特定的输入状态到输入映像表，最后继续执行剩余的程序，立即输入的示意图如图 1-14 所示。

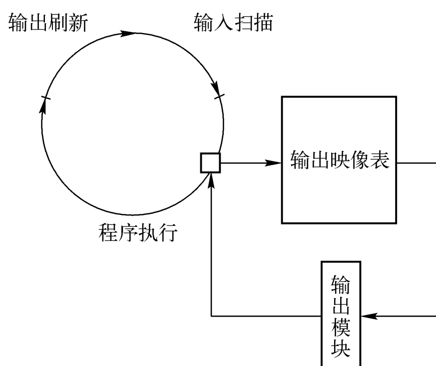


图 1-13　立即输出过程

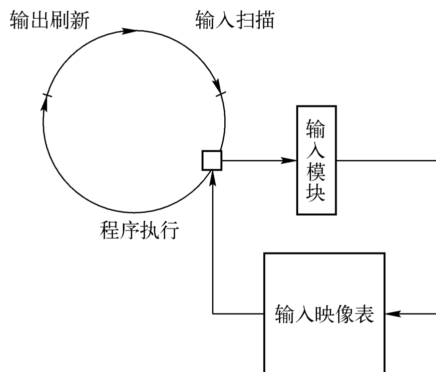
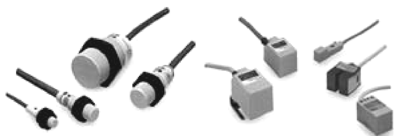


图 1-14　立即输入过程

1.3　接近开关

接近开关和 PLC 并无本质联系，但后续章节经常用到，所以以下将对此内容进行介绍。熟悉的读者可以跳过。

接近式位置开关是与（机器的）运动部件无机械接触而能操作的位置开关。当运动的物体靠近开关到一定位置时，开关发出信号，达到行程控制及计数自动控制。也就是说，它是一种非接触式无触点的位置开关，是一种开关型的传感器，简称接近开关（Proximity Sensors），又称接近传感器，外形如图 1-15 所示。接近开关有行程开关、微动开关的特性，又有传感性能，而且动作可靠，性能稳定，频率响应快，使用寿命长，抗干扰能力强等。它由感应头、高频振荡器、放大器和外壳组成。常见的接近开关有 LJ、CJ 和 SJ 等系列产品。接近开关的外形如图 1-15 所示，其图形符号如图 1-16a 所示，图 1-16b 所示为接近开关文字符号，表明接近开关为电容式接近开关，在画图时更加适用。


图 1-15　接近开关

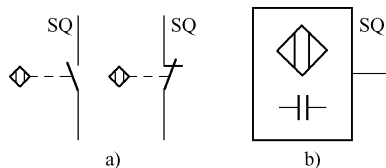


图 1-16　接近开关的图形及文字符号

1.3.1　接近开关的功能

当运动部件与接近开关的感应头接近时，就使其输出一个电信号。接近开关在电路中的作用与行程开关相同，都是位置开关，起限位作用，但两者是有区别的：行程开关有触点，是接触式的位置开关；而接近开关是无触点的，是非接触式的位置开关。

1.3.2　接近开关的分类和工作原理

按照工作原理区分，接近开关分为电感式、电容式、光电式和磁感式等形式。另外，根据应用电路电流的类型分为交流型和直流型。

① 电感式接近开关的感应头是一个具有铁氧体磁心的电感线圈，只能用于检测金属体，

在工业中应用非常广泛。振荡器在感应头表面产生一个交变磁场，当金属块接近感应头时，金属中产生的涡流吸收了振荡的能量，使振荡减弱以至停振，因而产生振荡和停振两种信号，经整形放大器转换成二进制的开关信号，从而起到“开”、“关”的控制作用。通常把接近开关刚好动作时感应头与检测物体之间的距离称为动作距离。

② 电容式接近开关的感应头是一个圆形平板电极，与振荡电路的地线形成一个分布电容，当有导体或其他介质接近感应头时，电容量增大而使振荡器停振，经整形放大器输出电信号。电容式接近开关既能检测金属，又能检测非金属及液体。电容式传感器体积较大，而且价格要贵一些。

③ 磁感式接近开关主要指霍尔接近开关，霍尔接近开关的工作原理是霍尔效应，当带磁性的物体靠近霍尔开关时，霍尔接近开关的状态翻转（如由“ON”变为“OFF”）。有的资料上将干簧继电器也归类为磁感式接近开关。

④ 光电式传感器是根据投光器发出的光，在检测体上发生光量增减，用光电变换元件组成的受光器检测物体有无、大小的非接触式控制器件。光电式传感器的种类很多，按照其输出信号的形式，可以分为模拟式、数字式、开关量输出式。

利用光电效应制成的传感器称为光电式传感器。光电式传感器的种类很多，其中，输出形式为开关量的传感器为光电式接近开关。

光电式接近开关主要由光发射器和光接收器组成。光发射器用于发射红外光或可见光。光接收器用于接收发射器发射的光，并将光信号转换成电信号，以开关量形式输出。

按照接收器接收光的方式不同，光电式接近开关可以分为对射式、反射式和漫射式 3 种。光发射器和光接收器有一体式和分体式两种形式。

⑤ 还有特殊种类的接近开关，如光纤接近开关和气动接近开关。特别是光纤接近开关在工业上使用越来越多，它非常适合在狭小的空间、恶劣的工作环境（高温、潮湿和干扰大）、易爆环境、精度要求高等条件下使用。光纤接近开关的问题是价格相对较高。

1.3.3 接近开关的选型

常用的电感式接近开关（Inductive Sensor）型号有 LJ 系列产品，电容式接近开关（Capacitive Sensor）型号有 CJ 系列产品，磁感式接近开关有 HJ 系列产品，光电型接近开关有 OJ 系列。当然，还有很多厂家都有自己的产品系列，一般接近开关型号的含义如图 1-17 所示。接近开关的选择要遵循以下原则。

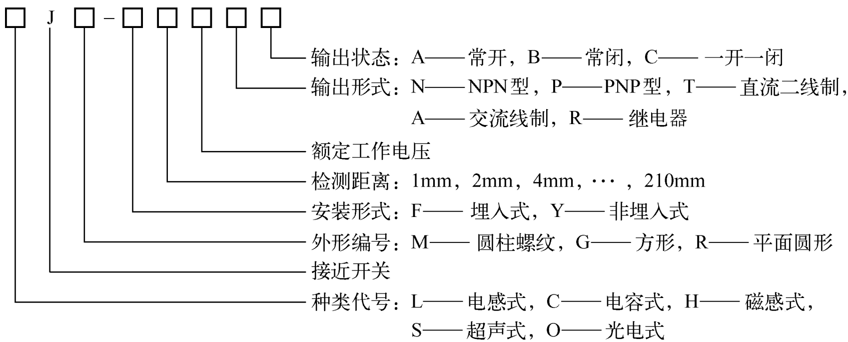


图 1-17　接近开关型号的含义

① 接近开关类型的选择。检测金属时优先选用电感式接近开关，检测非金属时选用电容式接近开关，检测磁信号时选用磁感式接近开关。

② 外观的选择。根据实际情况选用，但圆柱螺纹形状的最为常见。

③ 检测距离（Sensing Range）的选择。根据需要选用，但注意同一接近开关检测距离并非恒定，接近开关的检测距离与被检测物体的材料、尺寸以及物体的移动方向有关。表 1-3 列出了目标物体材料对检测距离的影响。不难发现，电感式接近开关对于有色金属的检测明显不如检测钢和铸铁。常用的金属材料不影响电容式接近开关的检测距离。

表 1-3 目标物体材料对检测距离的影响

序 号	目标物体材料	影响系数	
		电感式	电容式
1	碳素钢	1	1
2	铸铁	1.1	1
3	铝箔	0.9	1
4	不锈钢	0.7	1
5	黄铜	0.4	1
6	铝	0.35	1
7	紫铜	0.3	1
8	水	0	0.9
9	PVC（聚氯乙烯）	0	0.5
10	玻璃	0	0.5

目标的尺寸同样对检测距离有影响。满足以下任一条件时，检测距离不受影响。

- 当检测距离的 3 倍大于接近开关感应头的直径，而且目标物体的尺寸大于或等于 3 倍的检测距离 ×3 倍的检测距离（长 × 宽）。
- 当检测距离的 3 倍小于接近开关感应头的直径，而且目标物体的尺寸大于或等于检测距离 × 检测距离（长 × 宽）。

如果目标物体的面积达不到推荐数值时，接近开关的有效检测距离将按照表 1-4 推荐的数值减少。

表 1-4 目标物体的面积对检测距离的影响

占推荐目标面积的比例	影响系数	占推荐目标面积的比例	影响系数
75%	0.95	25%	0.85
50%	0.90		

④ 信号的输出选择。交流接近开关输出交流信号，而直流接近开关输出直流信号。注意，负载的电流一定要小于接近开关的输出电流，否则应添加转换电路解决。接近开关的信号输出能力见表 1-5。

⑤ 触点数量的选择。接近开关有常开触点和常闭触点。可根据具体情况选用。

表 1-5　接近开关的信号输出能力

接近开关种类	输出电流/mA	接近开关种类	输出电流/mA
直流二线制	50～100	直流三线制	150～200
交流二线制	200～350		

⑥ 开关频率的确定。开关频率是指接近开关每秒从“开”到“关”转换的次数。直流接近开关可达 200 Hz；而交流接近开关要小一些，只能达到 25 Hz。

⑦ 额定电压的选择。对于交流型的接近开关，优先选用 AC 220 V 和 AC 36 V，而对于直流型的接近开关，优先选用 DC 12 V 和 DC 24 V。

【例 1-2】 有人说：电感式接近开关，只能检测钢铁，不能检测其他金属。请问这句话对吗？

【解】

电感式接近开关，不但能检测钢铁，而且能检测其他金属，只是对铝和铜的检测灵敏度不如钢铁。

1.3.4　应用接近开关的注意事项

1. 单个 NPN 型和 PNP 型接近开关的接线

在直流电路中使用的接近开关有二线式（2 根导线）、三线式（3 根导线）和四线式（4 根导线）等多种，二线、三线、四线式接近开关都有 NPN 型和 PNP 型两种，通常日本和美国多使用 NPN 型接近开关，欧洲多使用 PNP 型接近开关，而我国则二者都有应用。NPN 型和 PNP 型接近开关的接线方法不同，正确使用接近开关的关键就是正确接线，这一点至关重要。

接近开关的导线有多种颜色，一般地，BN 表示棕色的导线，BU 表示蓝色的导线，BK 表示黑色的导线，WH 表示白色的导线，GR 表示灰色的导线，根据国家标准，各颜色导线的作用按照表 1-6 定义。对于二线式 NPN 型接近开关，棕色线与负载相连，蓝色线与零电位点相连；对于二线式 PNP 型接近开关，棕色线与高电位相连，负载的一端与接近开关的蓝色线相连，而负载的另一端与零电位点相连。图 1-18 和图 1-19 所示分别为二线式 NPN 型接近开关接线图和二线式 PNP 型接近开关接线图。

表 1-6　接近开关的导线颜色定义

种　　类	功　　能	接 线 颜 色	端　子　号
交流二线式和直流二线式（不分极性）	NO（接通）	不分正负极，颜色任选，但不能为黄色、绿色或者黄绿双色	3、4
	NC（分断）		1、2
直流二线式（分极性）	NO（接通）	正极棕色，负极蓝色	1、4
	NC（分断）	正极棕色，负极蓝色	1、2
直流三线式（分极性）	NO（接通）	正极棕色，负极蓝色，输出黑色	1、3、4
	NC（分断）	正极棕色，负极蓝色，输出黑色	1、3、2
直流四线式（分极性）	正极	棕色	1
	负极	蓝色	3
	NO 输出	黑色	4
	NC 输出	白色	2

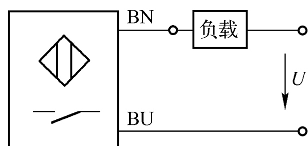


图 1-18　二线式 NPN 型接近开关接线图

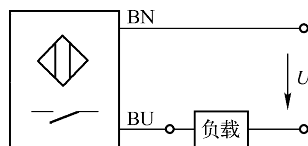


图 1-19　二线式 PNP 型接近开关接线图

表 1-6 中的“NO”表示常开、输出，而“NC”表示常闭、输出。

对于三线式 NPN 型接近开关，棕色的导线与一端负载，同时与电源正极相连；黑色的导线是信号线，与负载的另一端相连；蓝色的导线与电源负极相连。对于三线式 PNP 型接近开关，棕色的导线与电源正极相连；黑色的导线是信号线，与负载的一端相连；蓝色的导线与负载的另一端及电源负极相连，如图 1-20 和图 1-21 所示。

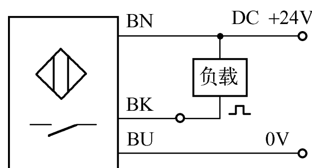


图 1-20　三线式 NPN 型接近开关接线图

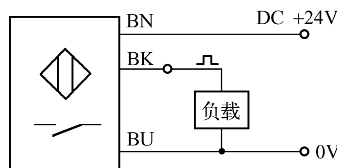


图 1-21　三线式 PNP 型接近开关接线图

四线式接近开关的接线方法与三线式接近开关类似，只不过，四线式接近开关多了一对触点而已，其接线图如图 1-22 和图 1-23 所示。

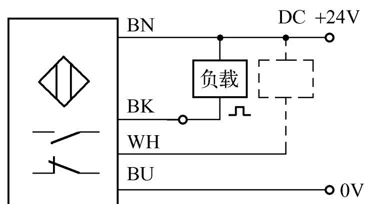


图 1-22　四线式 NPN 型接近开关接线图

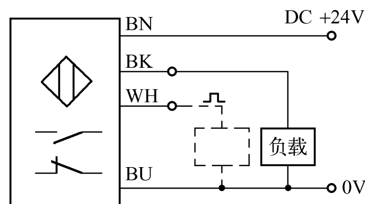


图 1-23　四线式 PNP 型接近开关接线图

2. 单个 NPN 型和 PNP 型接近开关的接线常识

初学者经常不能正确区分 NPN 型和 PNP 型的接近开关，其实只要记住一点：PNP 型接近开关是正极开关，也就是信号从接近开关流向负载；而 NPN 型接近开关是负极开关，也就是信号从负载流向接近开关。

【例 1-3】 在图 1-24 中，有一只 NPN 型接近开关与指示灯相连，当一个铁块靠近接近开关时，回路中的电流会怎样变化？

【解】

指示灯就是负载，当铁块到达接近开关的感应区时，回路突然接通，指示灯由暗变亮，电流从很小变化到 100% 的幅度，电流变化曲线如图 1-25 所示（理想状况）。

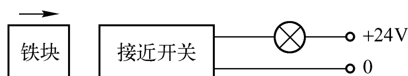


图 1-24　接近开关与指示灯相连的示意图

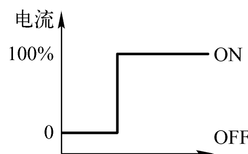


图 1-25　回路电流变化曲线

【**例 1-4**】某设备用于检测 PVC 物块，当检测物块时，设备上的 DC 24 V 功率为 12 W 的报警灯亮，请选用合适的接近开关，并画出原理图。

【**解**】

因为检测物体的材料是 PVC，所以不能选用电感接近开关，但可选用电容式接近开关。报警灯的额定电流为：$I_N=\frac{P}{U}=\frac{12}{24}A=0.5A$，查表 1-6 可知，直流接近开关承受的最大电流为 0.2 A，所以采用图 1-21 的方案不可行，信号必须进行转换，原理图如图 1-26 所示，当物块靠近接近开关时，黑色的信号线上产生高电平，其负载继电器 KA 的线圈得电，中间继电器 KA 的常开触点闭合，所以报警灯 EL 亮。

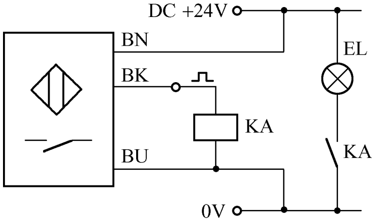


图 1-26　原理图

由于没有特殊规定，所以 PNP 或 NPN 型接近开关以及二线或三线式接近开关都可以选用。本例选用三线式 PNP 型接近开关。

3. 接近开关的并联

直流型接近开关允许并联接法，但一般不推荐将交流型接近开关进行并联。接近开关并联接线图如图 1-27 所示。二极管的作用是为了防止过载。当剩余电流足够小时，可以最多并联 30 个接近开关。

4. 接近开关的串联

接近开关串联接线图如图 1-28 所示，接近开关的数量取决于串联的接近开关的电压降低数，并与负载的要求电压密切相关。注意，当接近开关不适合或者不允许采用并联或者串联方式时，可以采用其他的电路实现同样的功能。

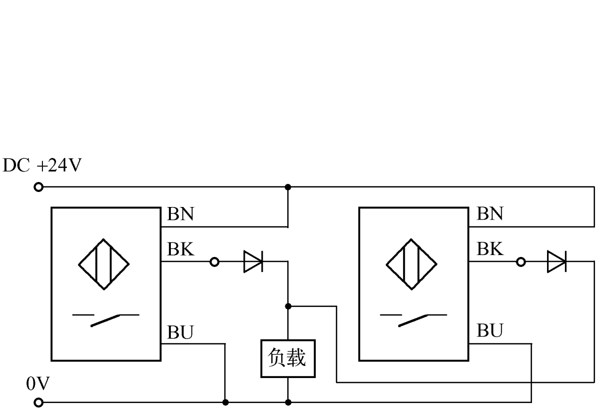


图 1-27　接近开关并联接线图

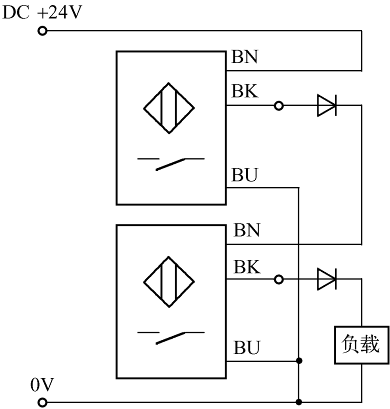


图 1-28　接近开关串联接线图

小　　结

（1）了解 PLC 的应用范围。
（2）重点掌握 PLC 的工作原理和结构。

习　　题

1. PLC 的主要性能指标有哪些？
2. PLC 主要用在哪些场合？
3. PLC 是怎样分类的？
4. PLC 的发展趋势是什么？
5. PLC 的结构主要有哪几个部分组成？
6. PLC 的输入和输出模块主要有哪几个部分组成？每部分的作用是什么？
7. PLC 的存储器可以细分为哪几个部分？
8. PLC 是怎样进行工作的？
9. 举例说明常见的哪些设备可以作为 PLC 的输入设备和输出设备？
10. 什么是立即输入和立即输出？在何种场合应用？
11. S7 系列的 PLC 有哪几类？
12. PLC 控制与继电器控制有何优缺点？
13. 三线式 NPN 型接近开关怎样接线？
14. 电容式和电感式开关传感器的区别是什么？
15. 两线/三线式 NPN 和 PNP 型接近开关怎样接线？
16. PLC 是什么控制系统基础上发展起来？(　　)
　　A. 继电控制系统　　B. 单片机　　C. 工业计算机　　D. 机器人

17、工业中控制电压一般是多少伏？(　　)
　　A. 24 V　　B. 36 V　　C. 110 V　　D. 220 V

18、工业中控制电压一般是（　　）。
　　A. 交流　　B. 直流　　C. 混合式　　D. 交变电压

19、请写出电磁兼容性英文缩写是（　　）。
　　A. MAC　　B. EMC　　C. CME　　D. AMC

第 2 章　S7 - 200 系列 PLC

本章主要介绍 S7 - 200 PLC 的 CPU 模块及其扩展模块的技术性能和接线方法；电源的需求计算。本章的内容非常重要。

2.1　S7 - 200 系列 PLC 简介

在 S7 - 200 系列 PLC 中，共有 11 个型号的 CPU，使用方法基本相同。

2.1.1　西门子 S7 系列模块简介

德国的西门子（SIEMENS）公司是欧洲最大的电子和电气设备制造商之一，生产的 SIMATIC可编程序控制器在欧洲处于领先地位。其第一代可编程序控制器是 1975 年投放市场的 SIMATIC S3 系列的控制系统。之后在 1979 年，西门子公司将微处理器技术应用到可编程序控制器中，研制出了 SIMATIC S5 系列，取代了 S3 系列，目前 S5 系列产品仍然有小部分在工业现场使用，在 20 世纪末，西门子又在 S5 系列的基础上推出了 S7 系列产品。最新的 SIMATIC 产品为 SIMATIC S7 和 C7 等几大系列。C7 是基于 S7 - 300 系列 PLC 性能，同时集成了 HMI。

SIMATIC S7 系列产品分为通用逻辑模块（LOGO!）、S7 - 200 系列、S7 - 200 SMART 系列、S7 - 1200 系列、S7 - 300 系列、S7 - 400 系列和 S7 - 1500 系列七个产品系列。S7 - 200 是在西门子收购的小型 PLC 的基础上发展而来，因此其指令系统、程序结构和编程软件和 S7 - 300/400 有较大的区别，在西门子 PLC 产品系列中是一个特殊的产品。S7 - 200 SMART 是 S7 - 200 的升级版本，是西门子家族的新成员，于 2012 年 7 月发布，其绝大多数的指令和使用方法与 S7 - 200 类似，其编程软件也和 S7 - 200 的类似，而且在 S7 - 200 的运行的程序，大部分可以在 S7 - 200 SMART 中运行。S7 - 1200 系列是在 2009 年才推出的新型小型 PLC，定位于 S7 - 200 和 S7 - 300 产品之间。S7 - 300/400 是由西门子的 S5 系列发展而来，是西门子公司的最具竞争力的 PLC 产品。2013 年西门子公司又推出了新品 S7 - 1500 系列产品。西门子 PLC 产品系列的定位见表 2-1。

表 2-1　西门子 PLC 产品系列的定位

序号	控　制　器	定　　位	主要任务和性能特征
1	LOGO!	低端独立自动化系统中简单的开关量解决方案和智能逻辑控制器	简单自动化 作为时间继电器、计数器和辅助接触器的替代开关设备 模块化设计，柔性应用 有数字量、模拟量和通信模块 用户界面友好，配置简单 使用拖放功能和智能电路开发

（续）

序号	控　制　器	定　　位	主要任务和性能特征
2	S7－200	低端的离散自动化系统和独立自动化系统中使用的紧促型辑控制器模块	串行模块结构、模块化扩展 紧促设计，CPU 集成 I/O 实时处理能力，高速计数器和报警输入和中断 易学易用的软件 多种通信选项
3	S7－200 SMART	低端的离散自动化系统和独立自动化系统中使用的紧促型辑控制器模块，是 S7－200 的升级版本	串行模块结构、模块化扩展 紧促设计，CPU 集成 I/O 集成了 PROFINET 接口 实时处理能力，高速计数器和报警输入和中断 易学易用的软件 多种通信选项
4	S7－1200	低端的离散自动化系统和独立自动化系统中使用的小型控制器模块	可升级及灵活的设计 集成了 PROFINET 接口 集成了强大的计数、测量、闭环控制及运动控制功能 直观高效的 STEP 7 Basic 工程系统可以直接组态控制器和 HMI
5	S7－300	中端的离散自动化系统中使用的控制器模块	通用型应用和丰富的 CPU 模块种类 高性能 模块化设计，紧促设计 由于使用 MMC 存储程序和数据，系统免维护
6	S7－400	高端的离散和过程自动化系统中使用的控制器模块	特别高的通信和处理能力 定点加法或乘法的指令执行速度最快为 0.03 μs 大型 I/O 框架和最高 20 MB 的主内存 快速响应，实时性强，垂直集成 支持热插拔和在线 I/O 配置，避免重启 具备等时模式，可以通过 PROFIBUS 控制高速机器
7	S7－1500	中高端系统	S7－1500 控制器除了包含多种创新技术之外，还设定了新标准，最大程度提高生产效率。无论是小型设备还是对速度和准确性要求较高的复杂设备装置，都一一适用。SIMATIC S7－1500 无缝集成到 TIA 博途中，极大提高了工程组态的效率

2.1.2　S7－200 PLC 的性能特点

SIMATIC S7－200 系列 PLC 适用于各行各业，各种场合中的检测、监测及控制的自动化。S7－200 系列的强大功能使其无论在独立运行中，或相连成网络皆能实现复杂控制功能。因此 S7－200 系列具有极高的性能价格比。从目前中国小型 PLC 的应用情况看，S7－200 的市场占有率占据领先地位，每年销售在 20 万台以上。

S7－200 系列出色表现在以下几个方面：

① 高的可靠性。

② 丰富的指令集。

③ 易于掌握。

④ 便捷的操作。

⑤ 丰富的内置集成功能。

⑥ 实时特性。

⑦ 强劲的通信能力。

⑧ 丰富的扩展模块。

2.2 S7-200 CPU 模块及其接线

2.2.1 S7-200 CPU 模块

S7-200 CPU 将微处理器、集成电源和多个数字量 I/O 点集成在一个紧凑的盒子中，形成功能比较强大的 S7-200 系列微型 PLC，如图 2-1 所示。

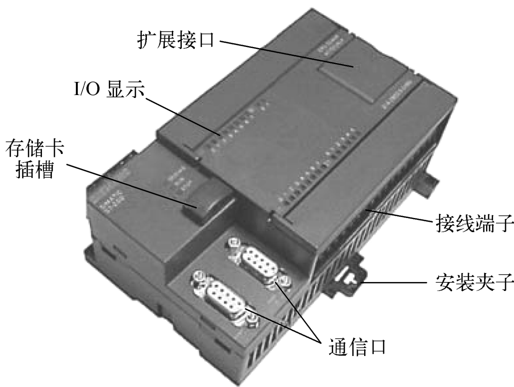


图 2-1 S7-200 PLC 外形

1. S7-200 CPU 的技术性能

西门子公司的 CPU 的中央处理器是 32 位的。西门子公司提供多种类型的 CPU，以适用各种应用要求，不同的 CPU 有不同的技术参数，其规格（节选）见表 2-2。读懂这个性能表是很重要的，设计者在选型时，必须要参考这个表格，例如晶体管输出时，输出电流为 0.75 A，若这个点控制一台电动机的起/停，设计者必须考虑这个电流是否足够驱动接触器，从而决定是否增加一个中间继电器。

表 2-2 S7-200 CPU 规格表

项目		CPU 221	CPU 222	CPU 224	CPU 224XP CPU 224XPSI	CPU 226
程序存储字节	使用运行编程模式	4096		8192	12288	16384
	不使用运行编程模式			12288	16384	24576
数字量 I/O		6/4	8/6	14/10		24/16
模拟量 I/O		无			2/1	无
本位通信口		1 个 RS-485		2 个 RS-485		
PPI、MPI/波特率		9.6、19.2、187.5 kbit/s				
自由口波特率		1.2～115.2 kbit/s				
高速脉冲输出/kHz		20×2			100×2	20×2

（续）

项　　目		CPU 221	CPU 222	CPU 224	CPU 224XP CPU 224XPSI	CPU 226
数字量输入特性		典型数值：DC 24 V，4 mA				
数字量输出特性		输出电压：DC 20.4 ~28.8 V 每个点的额定电流：0.75 A（晶体管输出）、2 A（继电器输出）				
供电能力/mA	DC 5 V	0	340	660		1 000
	DC 24 V	180	180	280		400
定时器		256				
计数器		256				

2. S7 –200 CPU 的工作方式

CPU 的前面板即存储卡插槽的上部，有 3 盏指示灯显示当前工作方式。指示灯为绿色时，表示运行状态；指示灯为红色时，表示停止状态；标有“SF”的灯亮表示系统故障，PLC 停止工作。

CPU 处于停止工作方式时，不执行程序。进行程序的上传和下载时，都应将 CPU 置于停止工作方式。停止方式可以通过 PLC 上的旋钮设定，也可以在编译软件中设定。

CPU 处于运行工作方式时，PLC 按照自己的工作方式运行用户程序。运行方式可以通过 PLC 上的旋钮设定，也可以在编译软件中设定。

2.2.2　S7 –200 CPU 的接线

1. CPU 22X 的输入端子的接线

S7 –200 系列 CPU 的输入端接线与三菱的 FX 系列的 PLC 的输入端接线不同，后者不需要接入直流电源，其电源由系统内部提供，而 S7 –200 系列 CPU 的输入端则必须接入直流电源。

下面以 CPU 224 为例介绍输入端的接线。“1 M”和“2 M”是输入端的公共端子，与 DC 24 V 电源相连，电源有两种连接方法对应 PLC 的 NPN 型（也称为漏型）和 PNP 型（也称源型）接法。当电源的负极与公共端子相连时，为 PNP 型接法，如图 2–2 所示；而当电源的正极与公共端子相连时，为 NPN 型接法，如图 2–3 所示。“M”和“L +”端子可以向传感器提供 DC 24 V 的电压，注意这对端子不是电源输入端子。

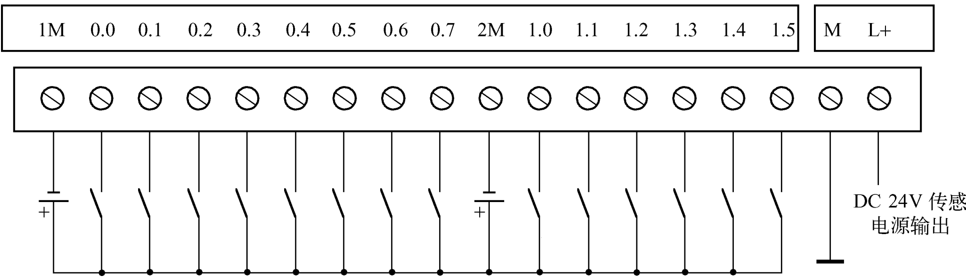


图 2–2　输入端子的接线（PNP）

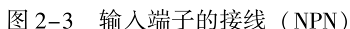


图 2-3　输入端子的接线（NPN）

初学者往往不容易区分 PNP 型和 NPN 型的接法，经常混淆，若读者记住以下的方法，就不会出错：把 PLC 作为负载，以输入开关（通常为接近开关）为对象，若信号从开关流出（信号从开关流出，向 PLC 流入），则 PLC 的输入为 PNP 型接法；把 PLC 作为负载，以输入开关（通常为接近开关）为对象，若信号从开关流入（信号从 PLC 流出，向开关流入），则 PLC 的输入为 NPN 型接法。三菱的 FX 系列（FX3U 除外）PLC 只支持 NPN 型接法。

【关键点】CPU 模块的高速输入（I0.3/4/5）可接收 DC 5 V 信号，其他输入点只可以接 DC 24 V 信号，只需将两种信号供电电源的公共端都连接到 1M 端子。但这两种信号必须同时为漏型或源型输入信号。

还有一点要指出，本书所指的 PNP 型和 NPN 型，对象为接近开关（传感器），这样便于与接近开关统一起来，很多大的 PLC 厂商都以这种方法讲解。而西门子 S7－200 手册上的讲解刚好与本书相反，原因在于 S7－200 手册上的 PNP 型（或者 NPN 型）是以 PLC 为对象说明。请读者务必注意这一点。

【例 2-1】有一台 CPU 224，输入端有一只三线 PNP 接近开关和一只二线 PNP 接近开关，应如何接线？

【解】

对于 CPU 224，公共端接电源的负极。而对于三线 PNP 接近开关，只要将其正、负极分别与电源的正、负极相连，将信号线与 PLC 的“I0.0”相连即可；而对于二线 PNP 接近开关，只要将电源的正极分别与其正极相连，将信号线与 PLC 的“I0.1”相连即可，如图 2-4 所示。

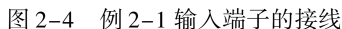


图 2-4　例 2-1 输入端子的接线

2. CPU 22X 的输出端子的接线

S7－200 系列 CPU 模块的数字量输出有两种形式，一种是 24 V 直流输出（即晶体管输出），另一种是继电器输出。PLC 上的标注“DC/DC/DC”的含义是：第一个 DC 的含义是供电电源电压 DC 24 V，第二个 DC 的含义是输入端的电源电压 DC 24 V，第三个 DC 的含义是输出为直流输出（晶体管输出）。对于西门子 S7－200 系列 CPU 模块，供电为 DC 24 V 的 CPU 模块，其输出为晶体管输出。“AC/DC/Relay”的含义是：AC 的表示供电电源电压 AC 220 V，DC 的表示输入端的电源电压 DC 24 V，Relay 的含义是输出为继电器输出。对于西门子 S7－200 系列

CPU 模块，供电为 AC 220 V 的 CPU 模块，其输出为继电器输出。

早期 24 V 直流输出只有一种形式，即 PNP 型输出，也就是常说的高电平输出，这点与三菱 FX 系列 PLC 不同，三菱 FX 系列 PLC（FX3U 除外，FX3U 有 PNP 型和 NPN 型两种可选择的输出形式）为 NPN 型输出，也就是低电平输出，理解这一点十分重要，特别是利用 PLC 进行运动控制（如控制步进电动机时）时，必须考虑这一点。后来，西门子也推出了 NPN 输出的 CPU 模块，即 CPU 224XPsi。

晶体管输出如图 2-5 所示。继电器输出没有方向性，可以是交流信号，也可以是直流信号，但不能使用 380 V 的交流电。继电器输出如图 2-6 所示。可以看出，输出是分组安排的，每组既可以是直流，也可以是交流电源，而且每组电源的电压大小可以不同，接直流电源时，没有方向性。在接线时，务必看清接线图。注意，当 CPU 的高速输出点 Q0.0 和 Q0.1 接 5 V 电源，其他点（如 Q0.2/3/4）接 24 V 电压时必须成组连接相同的电压等级。

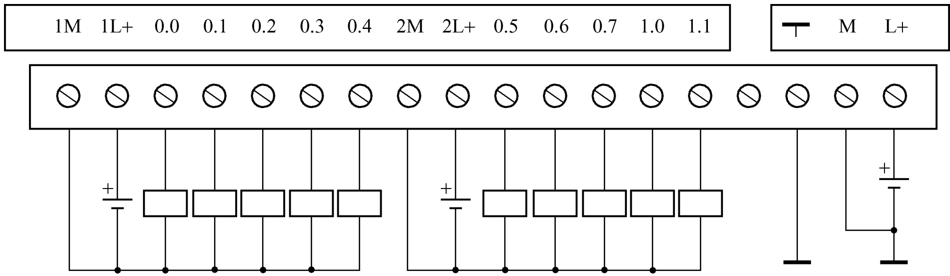


图 2-5　晶体管输出（PNP 型）

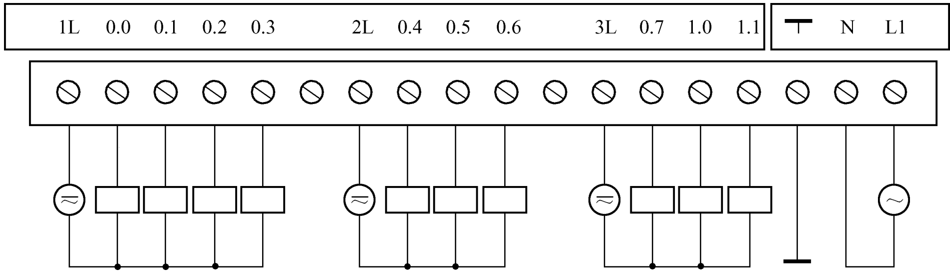


图 2-6　继电器输出

CPU 224XPsi 的输出比较特殊，是 NPN 型输出，其接线如图 2-7 所示。这个接线和三菱 FX2N 系列 PLC 的接线类似。

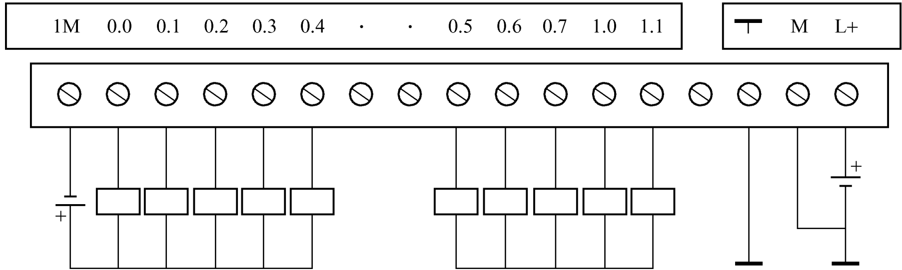


图 2-7　CPU 224XPsi 的输出接线图

在给 CPU 进行供电接线时，一定要特别小心分清是哪一种供电方式，如果把 AC 220 V 接到 DC 24 V 供电的 CPU 上，或者不小心接到 DC 24 V 传感器的输出电源上，都会造成 CPU 的损坏。

【例 2-2】 有一台 CPU 224，控制一只 DC 24 V 的电磁阀和一只 AC 220 V 电磁阀，输出端应如何接线？

【解】

因为两个电磁阀的线圈电压不同，而且有直流和交流两种电压，所以如果不经过转换，只能用继电器输出的 CPU，而且两个电磁阀分别在两个组中。其接线如图 2-8 所示。

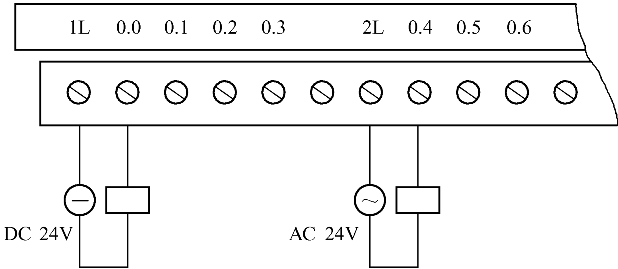


图 2-8　例 2-2 接线图

【例 2-3】 有一台 CPU 224，控制两台步进电动机和一台三相异步电动机的起/停，三相电动机的起/停由一只接触器控制，接触器的线圈电压为 AC 220 V，输出端应如何接线（步进电动机部分的接线可以省略）？

【解】 因为要控制两台步进电动机，所以要选用晶体管输出的 CPU，而且必须用 Q0.0 和 Q0.1 作为输出高速脉冲点控制步进电动机，但接触器的线圈电压为 AC 220 V，所以电路要经过转换，增加中间继电器 KA，其接线如图 2-9 所示。

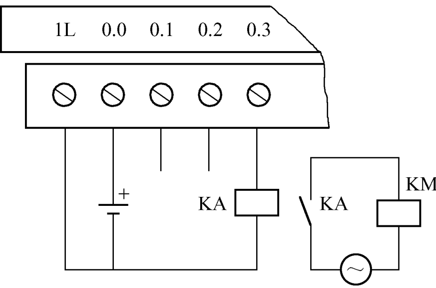


图 2-9　例 2-3 接线图

2.3　S7 -200 PLC 扩展模块

通常 S7 -200 系列 CPU 只有数字量输入和数字量输出（特殊除外，如 CPU 224XP），要完成模拟量的输入、模拟量输出、现场总线通信以及当数字输入输出点不够时，都应该选用扩展模块来解决。S7 -200 系列 PLC 有丰富的扩展模块供用户选用。S7 -200 的数字量、模拟量输入/输出模点不能复用（既能用做输入，又能用做输出）。

2.3.1　数字量 I/O 扩展模块

1. 数字量 I/O 扩展模块的规格

数字量 I/O 扩展模块包括数字量输入模块、数字量输出模块和数字量输入输出混合模块，当数字量输入或者输出点不够时可选用。部分数字量 I/O 模块的规格见表 2–3。

表 2–3　数字量 I/O 扩展模块规格表

型　　号	输　入　点	输　出　点	电　　压	功率/W	电 源 要 求	
					DC 5 V	DC 24 V
EM 221 DI	8	0	DC 24 V	1	30 mA	32 mA
EM 221 DI	8	0	AC 120/230 V	3	30 mA	—
EM 222 DO（DC 输出）	0	8	DC 24 V	3	50 mA	—
EM 222 DO（AC 输出）	0	8	DC 24 V	4	110 mA	—
EM 223（DC 输入、输出）	8	8	DC 24 V	2	80 mA	32 mA

2. 数字量 I/O 扩展模块的接线

数字量 I/O 模块有专用的扁平电缆与 CPU 通信，并通过此电缆由 CPU 向扩展 I/O 模块提供 DC 5 V 的电源。EM 221 数字量输入模块的接线如图 2–10 所示，EM 222 数字量输出模块的接线如图 2–11 所示。可以发现，数字量 I/O 扩展模块的接线与 CPU 的数字量输入输出端子的接线是类似的。

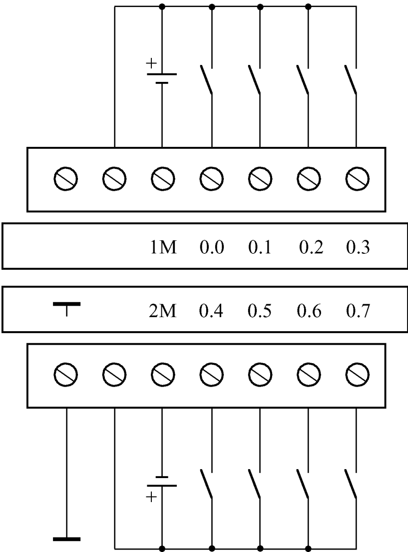


图 2–10　EM 221 模块接线图

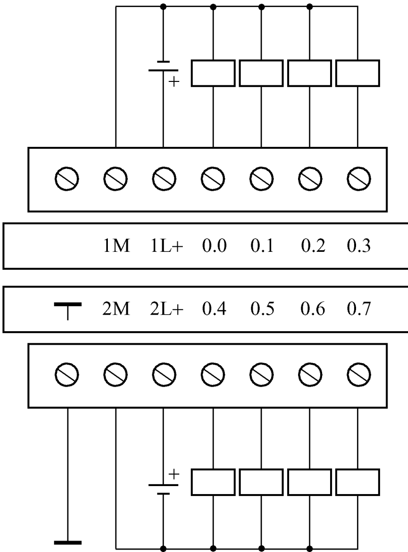


图 2–11　EM 222 模块接线图

S7 – 200 编程时不必配置 I/O 地址。S7 – 200 扩展模块上的 I/O 地址按照离 CPU 的距离递增排列。离 CPU 越近，地址号越小。在模块之间，数字量信号的地址总是以 8 位（1 个字节）为单位递增。如果 CPU 上的物理输入点没有完全占据一个字节，其中剩余未用的位

也不能分配给后续模块的同类信号。CPU 222（8 点输入、6 点输出）配置一块 EM 223（4 点输入、4 点输出）模块，扩展模块的输入地址为 I1.0 ~ I1.3，而 I1.4 ~ I1.7 空置不可用，扩展模块的输出地址为 Q1.0 ~ Q1.3，而 Q1.4 ~ Q1.7 空置不可用。

当 CPU 和数字量的扩展模块的输入点/输出点有信号输入或者输出时，LED 指示灯会亮，显示有输入/输出信号。

2.3.2 模拟量 I/O 扩展模块

1. 模拟量 I/O 扩展模块的规格

模拟量 I/O 扩展模块包括模拟量输入模块、模拟量输出模块和模拟量输入输出混合模块。部分模拟量 I/O 模块的规格见表 2-4。

表 2-4 模拟量 I/O 扩展模块规格表

型号	输入点	输出点	电压	功率/W	电源要求	
					DC 5 V	DC 24 V
EM 231	4	0	DC 24 V	2	20 mA	60 mA
	8	0	DC 24 V	2	20 mA	60 mA
EM 232	0	2	DC 24 V	2	20 mA	70 mA
	0	4	DC 24 V	2	20 mA	100 mA
EM 235	4	1	DC 24 V	2	30 mA	60 mA

2. 模拟量 I/O 扩展模块的接线

S7 - 200 系列的模拟量模块用于输入/输出电流或者电压信号。模拟量输入模块的接线如图 2-12 所示，模拟量输出模块的接线如图 2-13 所示。

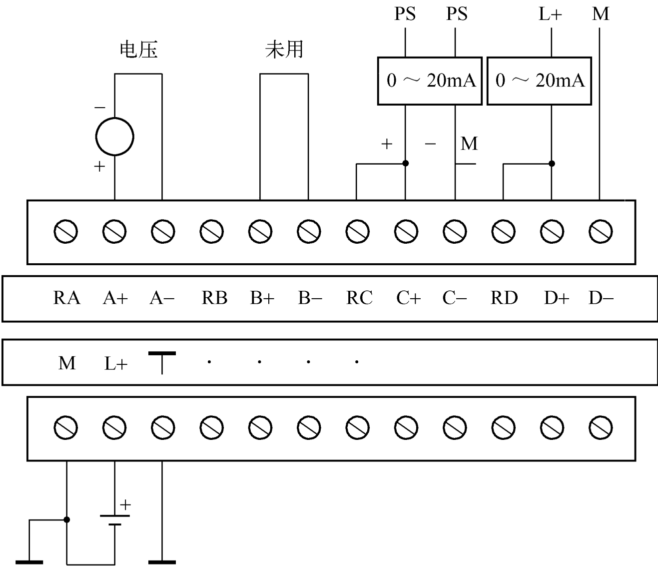


图 2-12 EM 231 模块接线图

虽然图 2-12 给出了模拟量输入信号的接法，但很多人在遇到具体传感器或者变送器时，仍然不会接线，因此以下对模拟量输入模块 EM 231 的接线分 4 种情况进行详细说明。

① 电压 4 线式接法。电压 4 线式接法如图 2-14 所示，这种接法最简单，一般的人都会。以 A 通道为例说明。RA 端子空置，传感器（或者变送器）的信号 + 与 A + 相连，传感器（或者变送器）的信号 - 与 A - 相连，传感器的电源 + 与电源的正极相连，传感器的电源 - 与电源的负极相连。

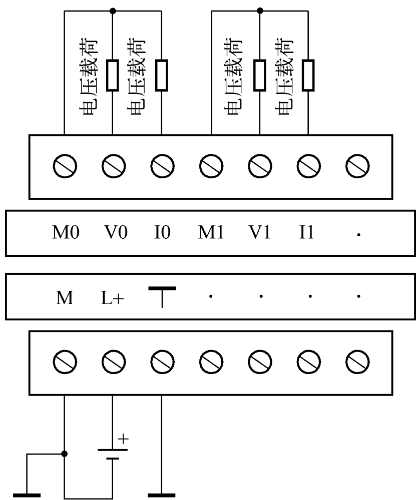


图 2-13　EM 232 模块接线图

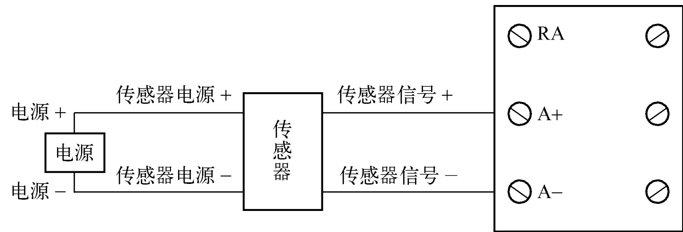


图 2-14　EM 231 模块接线图（电压 4 线式）

② 电流 4 线式接法。电流 4 线式接法如图 2-15 所示，这种接法也很简单，一般的人也会。以 A 通道为例说明。RA 端子与 A + 短接，传感器（或者变送器）的信号 + 与 A + 相连，传感器（或者变送器）的信号 - 与 A - 相连，传感器的电源 + 与电源的正极相连，传感器的电源 - 与电源的负极相连。

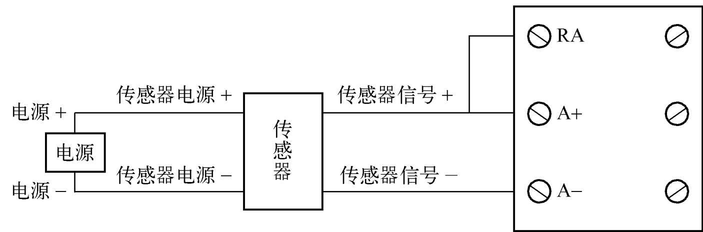


图 2-15　EM 231 模块接线图（电流 4 线式）

③ 电流 3 线式接法。电流 3 线式接法如图 2-16 所示。以 A 通道为例说明。RA 端子与 A + 短接，传感器（或者变送器）的信号 + 与 A + 相连，传感器（或者变送器）的信号 - 、

电源－与 A－相连。电流 3 线式的接法和电流 4 线式接法类似，只不过电流 3 线式的接法传感器的内部将传感器－和电源－短接在一起了。

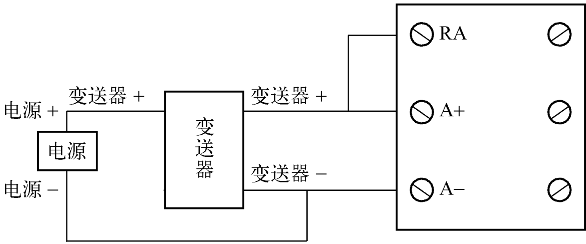


图 2-16　EM 231 模块接线图（电流 3 线式）

④ 电流 2 线式接法。电流 2 线式接法如图 2-17 所示，这种接法很多人不会接。以 A 通道为例说明。RA 端子与 A＋短接，传感器（或者变送器）的信号－与 A＋相连，传感器的信号＋与电源的正极相连，电源的负极与 A－相连。只要把变送器的回路当做串联回路看待，理解这种接线方法就不难了。

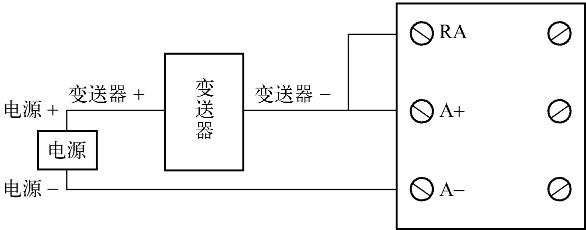


图 2-17　EM 231 模块接线图（电流 2 线式）

模拟量输入模块有两个参数容易混淆，即模拟量转换的分辨率和模拟量转换的精度（误差）。分辨率是 A－D 模拟量转换芯片的转换精度，即用多少位的数值来表示模拟量。若 S7－200 模拟量模块的转换分辨率是 12 位，能够反映模拟量变化的最小单位是满量程的 1/4 096。模拟量转换的精度除了取决于 A－D 转换的分辨率，还受到转换芯片的外围电路的影响。在实际应用中，输入的模拟量信号会有波动、噪声和干扰，内部模拟电路也会产生噪声、漂移，这些都会对转换的最后精度造成影响。这些因素造成的误差要大于 A－D 芯片的转换误差。

当模拟量的扩展模块的输入点/输出点有信号输入或者输出时，LED 指示灯不会亮，这点与数字量模块不同，因为西门子模拟量模块上的指示灯没有与电路相连。

使用模拟量模块时，要注意以下问题：

① 模拟量模块有专用的扁平电缆（与模块打包出售）与 CPU 通信，并通过此电缆由 CPU 向模拟量模块提供 DC 5 V 的电源。此外，模拟量模块必须外接 DC 24 V 电源。

② 每个模块能同时输入/输出电流或者电压信号，对于模拟量输入的电压或者电流信号选择通过 DIP 开关设定，量程的选择也是通过 DIP 开关来设定的。一个模块可以同时作为电流信号或者电压输入模块使用，但必须分别按照电流和电压型信号的要求接线。但是 DIP 开关设置对整个模块的所有通道有效，在这种情况下，电流、电压信号的规格必须能设置为相同的 DIP 开关状态。如表 2-5 中，0～5 V 和 0～20 mA 信号具有相同的 DIP 设置状态，可以

接入同一个模拟量模块的不同通道。

表 2-5　选择模拟量输入量程的 EM 231 配置开关表

	SW1	SW2	SW3	满　量　程	分　辨　率
单极性	ON	OFF	ON	0 ~ 10 V	2. 5 mV
		ON	OFF	0 ~ 5 V	1. 25 mV
				0 ~ 20 mA	5 μA
双极性	OFF	OFF	ON	±5 V	2. 5 mV
		ON	OFF	±2. 5 V	1. 25 mV

双极性就是信号在变化的过程中要经过“零”，单极性不过零。由于模拟量转换为数字量是有符号整数，所以双极性信号对应的数值会有负数。在 S7 – 200 PLC 中，单极性模拟量输入/输出信号的数值范围是 0 ~ 32 000；双极性模拟量信号的数值范围是 – 32 000 ~ 32 000。

③ 对于模拟量输入模块，传感器电缆线应尽可能短，而且应使用屏蔽双绞线，导线应避免弯成锐角。靠近信号源屏蔽线的屏蔽层应单端接地。

④ 未使用的通道应短接，如图 2-21 中的 B + 和 B – 端子未使用，进行了短接。

⑤ 一般电压信号比电流信号容易受干扰，应优先选用电流信号。电压型的模拟量信号由于输入端的内阻很高（S7 – 200 的模拟量模块为 10 MΩ），极易引入干扰。一般电压信号是用在控制设备柜内电位器设置，或者距离非常近、电磁环境好的场合。电流型信号不容易受到传输线沿途的电磁干扰，因而在工业现场获得广泛的应用。电流信号可以传输比电压信号远得多的距离。

⑥ 对于模拟量输出模块，电压型和电流型信号的输出信号的接线不同，各自的负载接到各自的端子上。

⑦ 前述的 CPU 和扩展模块的数字量的输入点和输出点都有隔离保护，但模拟量的输入和输出则没有隔离。如果用户的系统中需要隔离，请另行购买信号隔离器件。

⑧ 模拟量输入模块的电源地和传感器的信号地必须连接（工作接地），否则将会产生一个很高的上下振动的共模电压，影响模拟量输入值，测量结果可能是一个变动很大的不稳定的值。

⑨ 模拟量输出模块总是要占据两个通道的输出地址。即便有些模块（EM 235）只有一个实际输出通道，它也要占用两个通道的地址。在计算机和 CPU 实际联机时，使用Micro/WIN 的菜单命令“PLC”→“信息”，可以查看 CPU 和扩展模块的实际 I/O 地址分配。

【例 2-4】 有一个系统，配置了一台 EM 231，用于测量压力，压力传感器变送器输出的是 0 ~ 20 mA 的信号，请画出接线图？

【解】

压力传感器上有两个接线端子，正接线端子和 + 24 V 电源相连，负接线端子和 RA 及 A + 相连，电源的 0 V 和 A – 相连。不用的通道要短接，B + 和 B – 端子短接，C + 和 C – 端子短接，D + 和 D – 端子短接，接线图如图 2-18 所示。此外要注意拨码开关的位置，由于是电流信号，SW1 应为“ON”状态，SW2 应为“ON”状态，SW3 应为“OFF”状态。

【关键点】 本接线图实际是电流二线式接法，工程应用中使用传感器时，一定要认真阅读传感器的说明书，并根据其说明书接线。

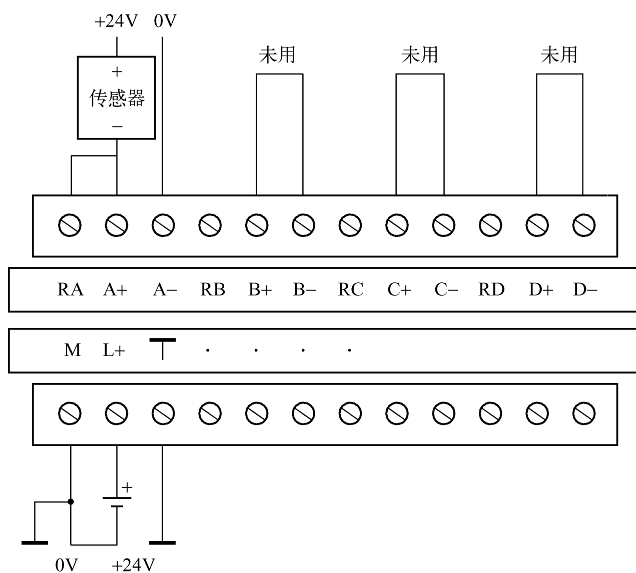


图 2-18　例 2-4 接线图

2.3.3　其他扩展模块

1. 热电偶、热电阻模块

EM 231 热电偶模块不同于常规的 EM 231 模拟量输入模块，它有冷端补偿电路，可以对测量数值作必要的修正，以补偿基准温度与模块温度差，同时，该模块的放大倍数较大，因此它能直接与热电偶相连，从而测量温度。EM 231 能和 J、K、E、N、S、T 及 R 七种热电偶相连，并测量温度。

EM 231 RTD 为热电阻模块，可以通过模块上的 DIP 开关选择热电阻的类型。

以下简要介绍 EM 231 热电偶模块。

EM 231 热电偶模块的接线如图 2-19 所示。

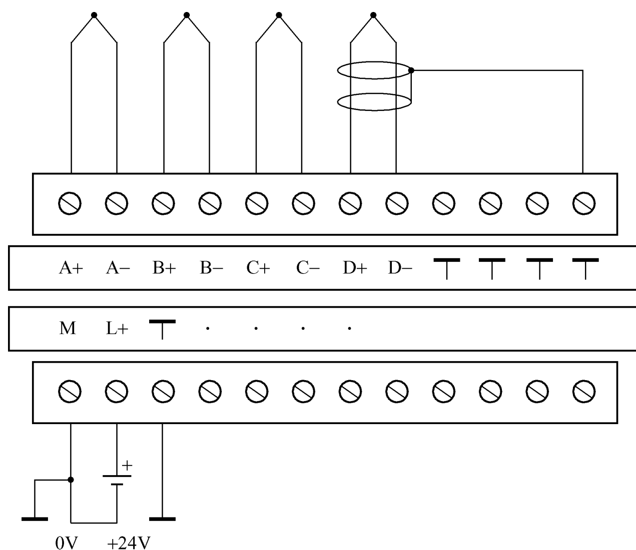


图 2-19　EM 231 热电偶模块的接线

EM 231 热电偶模拟量模块上有 8 个拨码开关，其中 4 不用，其余的开关都有特定的含义，在使用此模块时特别重要，其设置方法见表 2-6。

表 2-6　DIP 拨码开关的设置方法

序号	拨码开关位置	项目	设置	含　义
1	SW1、SW2、SW3	热电偶类型		通过改变 SW1、SW2、SW3 三个拨码的位置，选择适合连接热电偶的类型。例如，选用的热电偶是 K 型，那么 SW1 设为 0、SW2 设为 0、SW3 设为 1
	SW1,2,3 1 2 3 4 5 6 7 8 拨码开关向上表示接通，数值为 1；反之，拨码开关向下表示断开，数值为 0	J	000	
		K	001	
		T	010	
		E	011	
		R	100	
		S	101	
		N	110	
2	SW5	断线检测方向	设置	0 指示断线为正； 1 指示断线为负
	SW5 1 2 3 4 5 6 7 8	+32726. 7°	0	
		-32726. 7°	1	
3	SW6	断线检测启用	设置	将 25 μA 电流注入输入端子，可完成断线检测。断线检测启用开关可以启用或禁用检测电流。如果输入信号超出大约 200 mV，EM 231 热偶模块将检测断线。如检测到断线，测量读数被设定成由断线检测所选定的值
	SW6 1 2 3 4 5 6 7 8	启用	0	
		禁用	1	
4	SW7	温度选择	设置	设置为 0 表示获得数据为摄氏度； 设置为 1 表示获得数据为华氏度
	SW7 1 2 3 4 5 6 7 8	摄氏度	0	
		华氏度	1	
5	SW8	启用冷端补偿	设置	使用热电偶必须进行冷端补偿，如果没有启用冷端补偿，模块的转换则会出现错误。因为热电偶导线连接到模块连接器时会产生电压，选择 ±80 mV 范围时将自动禁用冷结点补偿
	SW8 1 2 3 4 5 6 7 8	启用	0	
		不启用	1	

【例 2-5】 有一台 EM 231 热电偶模块，用于测量温度，热电偶是 J 型，断线检测方向为正，启用断线检测，采用华氏温度，启用冷端补偿，请正确设置 DIP 拨码开关。

【解】

因为采用 J 型热电偶，SW1、SW2、SW3 设置均为 0；启用断线检测，且为正向 SW5、SW6 设置均为 0；为华氏温度，SW7 =1；启用冷端补偿 SW8 =0。其设置如图 2-20 所示。

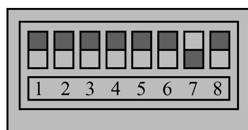


图 2-20　DIP 拨码开关设置图

2. 通信模块 PROFIBUS－DP（EM 277）

S7－200 系列的 CPU 要接入 PROFIBUS－DP 网，则必须配置 EM 277 模块，EM 277 作为 DP 从站（只能作为从站，S7－300 和 S7－400 可以作为主站），EM 277 模块接收来自主站的多种不同的 I/O 组态，向主站发送和接收数据。此外，EM 277 还可以作为 MPI 的从站，与主站（如 S7－300）进行 MPI 通信。正因为 EM 277 是 PROFIBUS－DP 从站模块，不能做主站，而西门子的变频器需要接受主站的控制，所以 EM 277 不能控制西门子变频器。S7－200 不能作为 PROFIBUS－DP 的主站。EM 277 模块并不占用地址。

当 CPU 上的通信口（如自由口通信等）已经被占用，或者 CPU 的连接数已经用尽时，如果还要连接 HMI，则可以在 CPU 上附加 EM 277 模块，EM 277 上的通信口可以连接西门子的 HMI。其他品牌的 HMI 是否能够连接要询问其生产厂家。注意，不能扩展出与 S7－200 CPU 通信口功能完全一样的通信口。

对 EM 277 重新设置地址后，须断电后重新上电才起作用。有时重新设置且断电后仍然不起作用，则要检查 EM 277 地址拨码是否到位。

通信模块中还有 Modem 模块 EM 241 和以太网模块 CP 243－1 等。

3. 定位模块（EM 253）

S7－200 系列 CPU（晶体管输出时）的 Q0.0 和 Q0.1 可以输出高速脉冲，可以用于控制步进电动机和伺服电动机，但若要求较高时，则应使用定位模块 EM 253。EM 253 占用输出 1QB 地址。

2.4　S7－200 PLC 的安装

S7－200 PLC 设备设计易于安装。S7－200 PLC 可采用水平或垂直方式安装在面板或标准 DIN 导轨上。S7－200 PLC 体积小，用户能更有效地利用空间。S7－200 PLC 安装的时候一般应遵循如下原则。

2.4.1　安装的预留空间

1. 将 S7－200 PLC 与热源、高电压和电子噪声隔离开

按照惯例，在安装元器件时，总是把产生高电压和高电子噪声的设备与诸如 S7－200 PLC 这样的低压、逻辑型的设备分隔开。

在控制柜背板上安排 S7－200 PLC 时，应区分发热装置并把电子器件安排在控制柜中温度较低的区域内。电子器件在高温环境下工作会缩短其无故障时间。还要考虑面板中设备的布线。避免将低压信号线和通信电缆与交流供电线和高能量、开关频率很高的直流线路布置在一个线槽中。

2. 为接线和散热留出适当的空间

S7－200 PLC 设备设计成通过自然对流冷却。为保证适当冷却，必须在设备上方和下方留出至少 25 mm 的间隙。此外，模块前端与机柜内壁间至少应留出 25 mm 的深度。预留空间参考如图 2-21 所示。

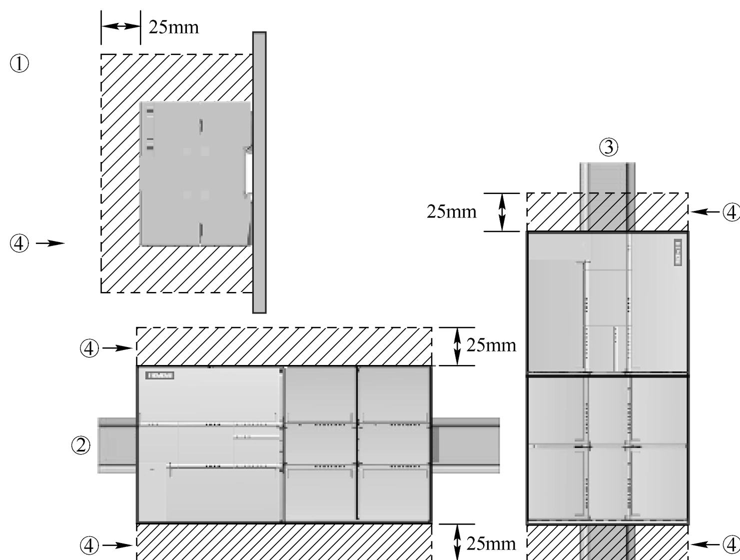


图 2-21　预留空间示意图

2.4.2　S7－200 PLC 模块的安装和拆卸

1. S7－200 PLC 模块的安装

S7－200 PLC 可以很容易地安装在一个标准 DIN 导轨或控制柜背板上。可使用 DIN 导轨卡夹将设备固定到 DIN 导轨上。具体步骤如下：

1）每隔 75 mm 将 DIN 导轨（35 mm）固定到安装板上。

2）咔嚓一声打开模块底部的 DIN 夹片，并将模块背面卡在 DIN 导轨上，如图 2-22 所示。

3）如果使用扩展模块，则将其置于 DIN 导轨上的 CPU 旁。

4）将模块向下旋转至 DIN 导轨，咔嚓一声闭合 DIN 夹片，如图 2-23 所示。仔细检查夹片是否将模块牢牢地固定到导轨上。为避免损坏模块，请按安装孔标记处而不要直接按模块前侧。

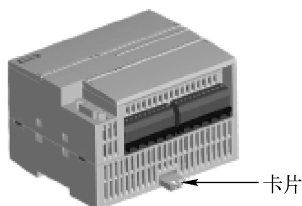


图 2-22　打开卡片

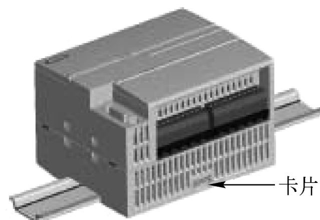


图 2-23　闭合卡片

2. S7 -200 PLC 模块的拆卸

按照以下步骤拆卸 S7 -200 CPU 或扩展模块：

1）切断电源，拆卸 S7 -200 PLC 的电源。

2）拆卸模块上的所有连线和电缆。大多数的 CPU 和扩展模块有可拆卸的端子排，使这项工作变得简单。

3）如果有其他扩展模块连接在所拆卸的模块上，打开盖板，拔掉相邻模块的扩展扁平电缆。

4）拆掉安装螺钉或者打开 DIN 夹子。

5）拆下模块。

3. 拆卸端子排

为了安装和替换模块方便，大多数的 S7 -200 PLC 模块都有可拆卸的端子排。目前只有 CPU221 和 CPU22 共 4 款（每种 2 款）CPU 的端子排不能拆卸，其余 7 款均可拆卸。也可以为固定端子排的模块订购可选的扇出连接排。

端子排的拆卸方法如下：

1）打开端子排安装位置的上盖板，以便可以接近端子排。

2）把螺钉旋具插入端子块中央的槽口中。

3）用力下压并撬出端子排，如图 2-24 所示。

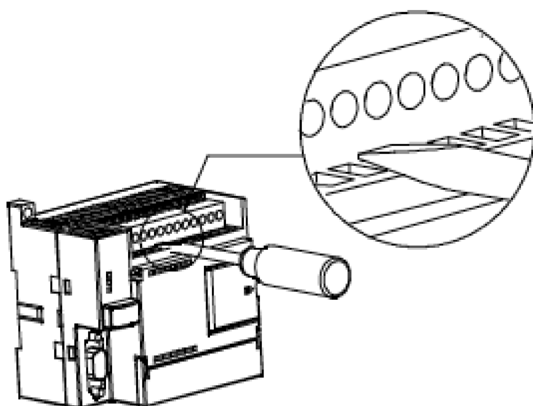


图 2-24 端子拆卸图

2.5 电源需求

2.5.1 最大 I/O 配置

1. 最大 I/O 的限制条件

① CPU 的 I/O 映像区的大小限制。

② CPU 本体的 I/O 点数的不同。

③ CPU 所能扩展的模块数目，如 CPU 224 能扩展 7 个模块，CPU 222 能扩展 2 个模块。

④ CPU 内部 +5 V 电源是否满足所有扩展模块的需要，扩展模块的 +5 V 电源不能外接

电源，只能由 CPU 供给。

⑤ CPU 智能模块对 I/O 点地址的占用。

而在以上因素中，CPU 的供电能力对扩展模块的个数起决定因素，因此最为关键。

2.5.2 电源需求计算

所谓电源计算，就是用 CPU 所能提供的电源容量减去各模块所需要的电源消耗量。S7－200 CPU 模块提供 DC 5 V 和 DC 24 V 电源。当有扩展模块时，CPU 通过 I/O 总线为其提供 5 V 电源，所有扩展模块的 5 V 电源消耗之和不能超过该 CPU 提供的电源额定值。若不够用，不能外接 5 V 电源。

每个 CPU 都有一个 DC 24 V 传感器电源，它为本机输入点和扩展模块输入点及扩展模块继电器线圈提供 DC 24 V。如果电源要求超出了 CPU 模块的电源定额，可以增加一个外部 DC 24 V 电源来供给扩展模块，但只能二选一，不能同时由外接电源和 CPU 同时供电。

注意，EM 277 模块本身不需要 DC 24 V 电源，这个电源是专供通信端口用的。DC 24 V 电源的需求取决于通信端口上的负载大小。CPU 上的通信口可以连接 PC/PPI 电缆和 TD 200 并为它们供电，此电源消耗已经不必再纳入计算。下面举例说明电源的需求计算。

【例 2-6】 某系统上有 1 个 CPU 224、1 个 EM 221 模块和 3 个 EM 223 模块，计算由 CPU 224 供电，电源是否足够？

【解】

首先查表 2-1 可知，CPU 224（AC/DC/继电器）的供电能力是 DC 5 V 电压时，提供最大 660 mA 电流和 DC 24 V 电压时，最大提供 280 mA 电流。

因为（660－1×30－3×80）mA＝390 mA，可见 DC 5 V 电源是足够的。

又因为（280－14×5－1×8×5－3×8×5－3×8×9）mA＝－120 mA，可见 DC 24 V 电源是不够的。

因此，DC 24 V 电源需要使用外加电源，但是注意外加电源不能与 CPU 模块本身的电源并联在一起，可直接连到需要的电源上。

小　　结

（1）S7－200 系列 PLC 的外部接线、扩展模块的接线，特别是数字量输入、输出模块和模拟量输入/输出模块的接线至关重要。

（2）电源的需求计算既是重点，也是难点，特别要学会通过产品手册查询相关参数。

习　　题

1. 举例说明常见的哪些设备可以作为 PLC 的输入设备和输出设备？
2. S7 系列的 PLC 有哪几类？
3. S7－200 系列 PLC 有什么特色？
4. S7－200 系列 CPU 有几种工作方式？下载文件时，能否使其置于“运行”状态？
5. 使用模拟量输入模块时，要注意什么问题？

6. 在例 2-2 中，如果不经过转换能否直接用晶体管输出 CPU 代替？应该如何转换？

7. 如何进行 S7-200 的电源需求与计算？

8. 是否可以通过 EM 277 模块控制变频器？

9. 为什么重新设置 EM 277 地址后不起作用？

10. S7-200 系列 PLC 的输入和输出怎样接线？

11. EM 231 热电偶模块的 DIP 开关 SW1～SW8 分别代表的含义是什么？

12. 某系统上有 1 个 CPU 226、2 个 EM 221 模块、3 个 EM 222 模块、1 个 EM 231 模块和 1 个 EM 232 模块，计算由 CPU 226 供电，电源是否足够？

13. PLC 自控系统中，温度控制，可用什么扩展模块？(　　)

A. EM 221　　B. EM 222　　C. EM 232　　D. EM 231 RTD

14. 西门子 S7-200 系列 PLC 最多可以有多少个点（含扩展）？(　　)

A. 168　　B. 128　　C. 256　　D. 188

15. 以下哪个 PLC 不具备扩展能力？(　　)

A. CPU 221　　B. CPU 222　　C. CPU 224XP　　D. CPU 226

第 3 章　西门子 S7 - 200 PLC 编程软件

本章主要介绍 STEP 7 - Micro/WIN 软件的安装和使用方法，以及仿真软件的使用，学会本章内容是编程必备的。

3.1　STEP 7 - Micro/WIN 编程软件的安装

3.1.1　STEP 7 - Micro/WIN 编程软件概述

STEP 7 - Micro/WIN 是一款功能比较强大的软件，此软件易学易用，用于 S7 - 200 系列 PLC 编程软件，支持 3 种模式：LAD（梯形图）、FBD（功能块图）和 STL（语句表）。STEP 7 - Micro/WIN 可提供程序的在线编辑、监控和调试。本书介绍的 STEP 7 - Micro/WIN V4.0 版本，可以兼容旧版本的程序，但新版本的程序不能在旧版本的软件上使用。

如果读者没有安装此软件，可以在西门子（中国）自动化与驱动集团的网站（http://www.ad.siemens.com.cn/）上下载软件安装使用。

安装此软件对计算机的要求如下：

① Windows 2000 SP3 以上操作系统，或者 Windows XP Home、Windows XP Professional 和 Windows 7 操作系统，但建议不要使用 Windows XP Home 操作系统，因为在此操作系统上有些功能受限，如不能使用 CP 卡通信。要注意在 Windows 7 上安装必须是 STEP 7 - Micro/WIN V4.0 SP9 版本，最好使用旗舰版的 Windows 7 操作系统。

② 至少 350 MB 硬盘空间。

有了 PLC 和配置必要软件的计算机，两者之间必须有一根编程电缆，此电缆与计算机端相连的一般是 RS - 232C 接口形式，与 PLC 端相连的是编程口（RS - 485 接口形式），有的 PLC 只有一个编程口（如 CPU 221），有的 PLC 只有两个编程口（如 CPU 226），任何一个编程口与编程电缆相连均可，其连接如图 3-1 所示。还有 USB 形式的编程电缆出售，但这种编程电缆不支持自由口协议（例如，不能用 Windows 下的超级终端与 PLC 利用自由口通信）。当然，也可以使用 CP 卡通信，CP 卡的功能比以上的编程电缆的功能强得多，它还可以用于 S7 - 300 和 S7 - 400 系列 PLC，但价格要贵很多。

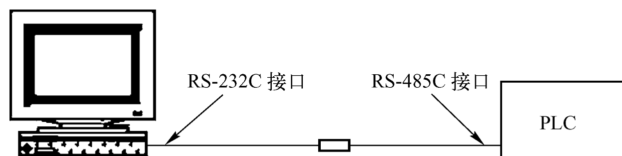


图 3-1　计算机的 RS - 232C 接口与 PLC 的连线图

【关键点】如果有的笔记本电脑没有 RS－232C 接口，并不意味着不能使用，读者只需要将 USB－RS232（USB/Serials）转换器安装在笔记本电脑的 USB 接口和通信电缆的 RS－232C 接口之间，即可实现可靠的通信。但对于有的计算机，可能会出现通信故障，请读者认真阅读 USB/Serials 转换器的说明书，并安装 USB/Serials 转换器的驱动程序。建议最好使用西门子的原装编程电缆，市场上的某些仿制电缆外观与西门子的电缆相似，但使用时有时会出现通信失败现象。

3.1.2 STEP 7－Micro/WIN 编程软件的安装步骤

STEP 7－Micro/WIN 编程软件的安装步骤如下：

1）打开 STEP 7－Micro/WIN 编程软件的安装包，双击可执行文件“SETUP. EXE”，软件安装开始，并弹出选择设置语言对话框，如图 3-2 所示，有 5 种语言可供选择，选择“英语”。此时弹出安装向导对话框，如图 3-3 所示，单击“Next”（下一步）按钮即可。之后弹出安装许可协议界面，如图 3-4 所示，单击“Yes”按钮，表示同意许可协议，否则安装不能继续进行。

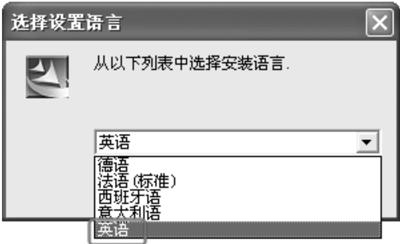


图 3-2 选择设置语言

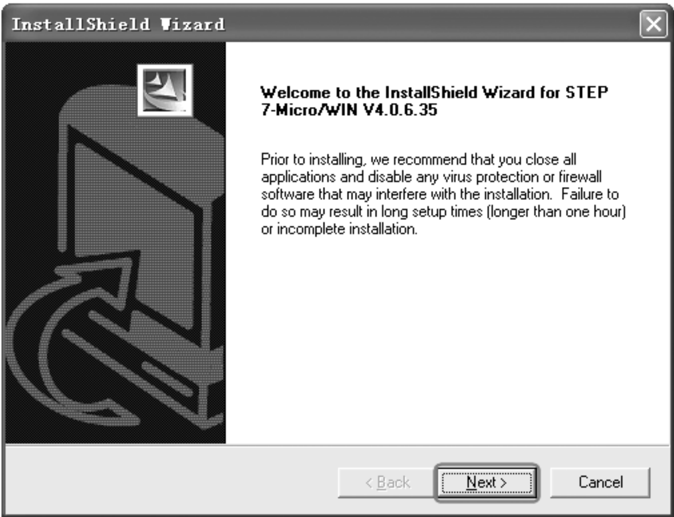


图 3-3 安装向导

2）选择安装目录。如果要改变安装目录，则单击“Browse”（浏览）按钮，选定想要安装的目录即可；如果不想改变目录，则单击“Next”（下一步）按钮，如图 3-5 所示，程序开始安装，并显示安装进程，如图 3-6 所示。

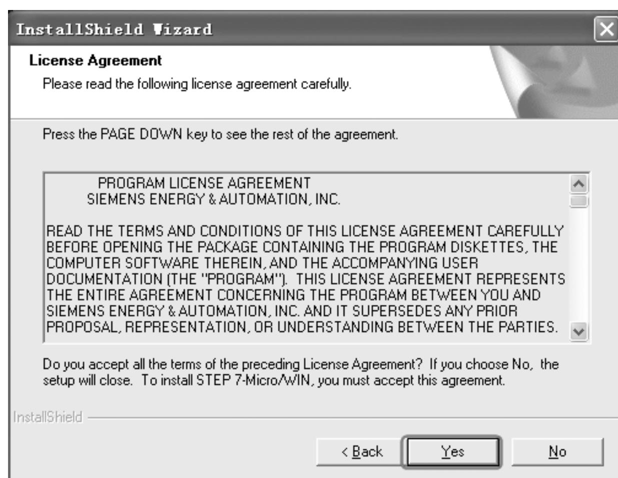


图 3-4 安装许可协议

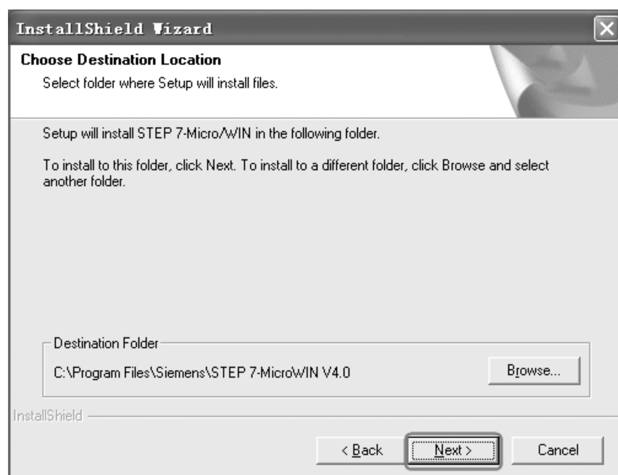


图 3-5　选择安装目录

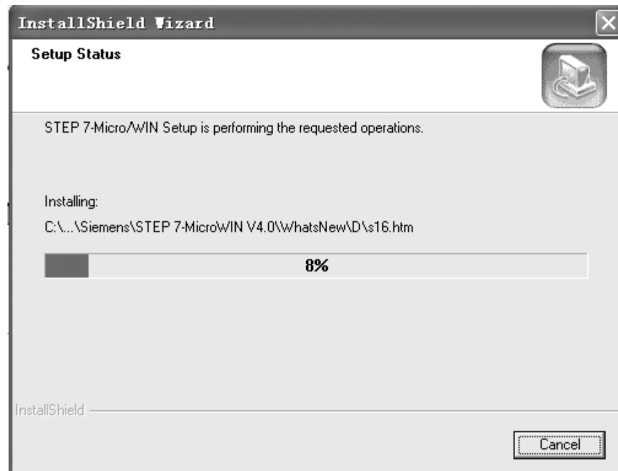


图 3-6　安装进程

3）选择默认的下载适配器。程序安装即将结束时，会弹出“设置 PG/PC Interface”，如图 3-7 所示，选中 PC/PPI cable（PPI），再单击“OK”（确定）按钮。

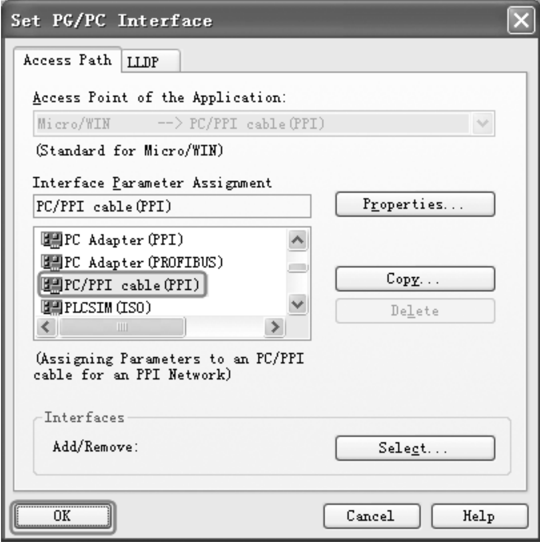


图 3-7　设置 PG/PC Interface

4）当软件安装结束时，弹出是否要重启计算机界面，如图 3-8 所示，建议选择“Yes, I want to restart my computer now.”（是，我想现在重启我的计算机），再单击“Finish”（完成）按钮，重启计算机后，再运行 STEP 7 - Micro/WIN 软件。尽管不重启计算机，也能运行 STEP 7 - Micro/WIN 软件，但仍然建议重启计算机。

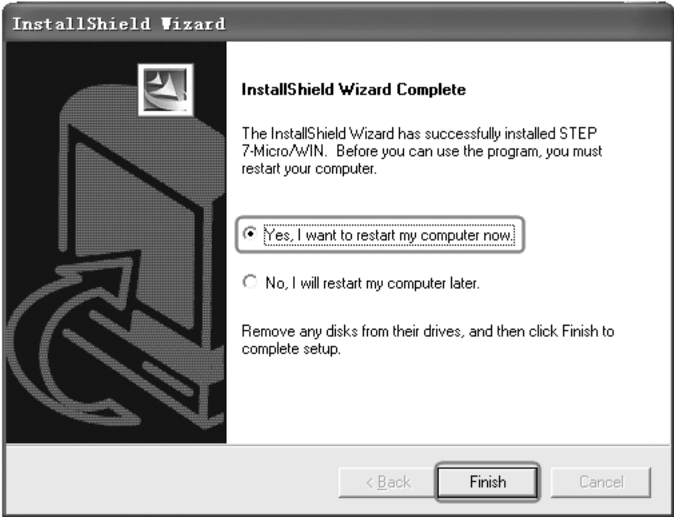


图 3-8　设置 PG/PC 界面

【关键点】 安装 STEP 7 - Micro/WIN 软件前，最好关闭杀毒和防火墙软件，此外存放 STEP 7 - Micro/WIN 软件的目录最好是英文。

3.2　STEP 7 – Micro/WIN 的使用

3.2.1　STEP 7 – Micro/WIN 软件的打开

打开 STEP 7 – Micro/WIN 软件通常有三种方法，分别介绍如下：

① 单击“所有程序”→“Simatic”→“STEP 7 – Micro/WIN V4.0.6.35”→“STEP 7 – Micro/WIN”，如图 3-9 所示，即可打开软件。

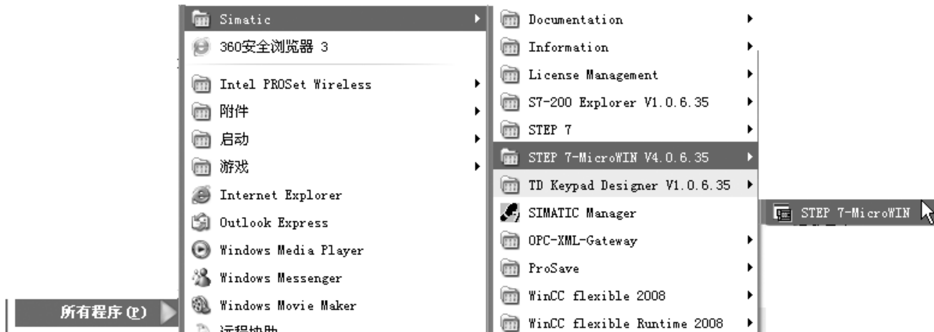


图 3-9　打开 STEP 7 – Micro/WIN 软件界面

② 直接双击桌面上的 STEP 7 – Micro/WIN 软件快捷方式，也可以打开软件，这是较快捷的打开方法。

③ 在电脑的任意位置，双击以前保存的程序，即可打开软件。

3.2.2　STEP 7 – Micro/WIN 软件的界面介绍

STEP 7 – Micro/WIN 软件的主界面如图 3-10 所示。其中包含菜单栏、工具浏览条、工具栏、指令树、程序编辑器、输出窗口等。

1. 菜单栏

菜单栏包括文件、编辑、查看、PLC、调试、工具、窗口和帮助 8 个菜单项。用户可以定制“工具”菜单，在该菜单中增加自己的工具。

2. 工具浏览条

工具浏览条显示编程特性的按钮控制群组。它在编译程序时是非常有用的，尽管其功能在菜单中同样可以实现，但显然使用工具浏览条更为方便。

工具浏览条中有“查看”和“工具”两个视图。“查看”视图显示了程序块、符号表、状态表、数据块、系统块、交叉引用及通信工具。“工具”视图显示了指令向导、文本显示向导、位置控制向导、EM 253 控制面板和调制解调器扩展向导等工具。工具浏览条的“工具”视图中的按钮功能与菜单栏中的“工具”菜单的功能相同。工具浏览条中还提供了滚动按钮，方便用户查看对象。

3. 指令树

指令树提供所有项目对象和为当前程序编辑器（LAD、FBD 或 STL）提供所有指令的树



图 3-10 STEP 7 - Micro/WIN 软件的主界面

形视图。用户可以右击指令树中的"项目"节点，插入附加程序组织单元（POU）；可以右击单个 POU，打开、删除、编辑其属性表，添加密码保护或重命名子程序及中断例行程序。可以右击指令树中"指令"节点或单个指令，以便隐藏整个树。展开指令树中的节点，可以拖放单个指令，或双击指令系统自动将所选指令插入程序编辑器中的光标位置。用户可以将指令拖放到"偏好"节点中，排列经常使用的指令。界面如图 3-10 所示，具体功能如下：

① 可借助交叉引用（Cross Reference，也称交叉参考）检视程序的交叉引用和组件使用信息。

② 可借助数据块显示和编辑数据块内容。

③ 可借助"状态表"窗口允许将程序的输入、输出结果或变量置入图表中，以便追踪其状态。可以建立多个状态图，以便从程序的不同部分监视组件。每个状态图在"状态表"窗口中都有自己的标签。

④ 符号表/全局变量表窗口允许分配和编辑全局符号（即可在任何 POU 中使用的符号值，不只是建立符号的 POU）。可以建立多个符号表。可在项目中增加一个 S7－200 系统符号预定义表。

⑤ 输出窗口在编译程序时提供信息。当输出窗口列出程序的错误信息时，双击错误信息，会在程序编辑器窗口中显示适当的网络。

⑥ 状态栏显示进行 STEP 7 - Micro/WIN 操作时的操作状态信息。

⑦ 程序编辑器窗口包含用于该项目的编辑器（LAD、FBD 或 STL）的局部变量表和程序视图。如果需要，拖动分割条，扩展程序视图，并覆盖局部变量表。若在主程序一节

（OB1）之外，建立子程序或中断例行程序时，标记出现在程序编辑器窗口的底部。可单击该标记，在子程序、中断和 OB1 之间移动。

⑧ 局部变量表包含读者对局部变量所作的赋值（即子程序和中断例行程序使用的变量）。在局部变量表中建立的变量使用暂时内存，地址赋值由系统处理，并且变量的使用仅限于建立此变量的 POU。

4. 工具栏

工具栏为常用的操作提供便利的访问。用户可以定制每个工具栏的内容和外观。

（1）标准工具栏

标准工具栏如图 3-11 所示。其中，“编译程序或数据块”按钮和“全部编译”按钮的区别是，前者是在任意一个激活窗口中编译程序块或数据块，是局部编译，而后者则是对程序、数据块和系统块的全部编译，建议多使用“全部编译”按钮。“上载”按钮是将项目从 PLC 上载至 STEP 7 - Micro/WIN（有的称为“上传”或“读入”），而“下载”按钮是将项目从 STEP 7 - Micro/WIN 下载至 PLC（有的软件也称为“写出”）。

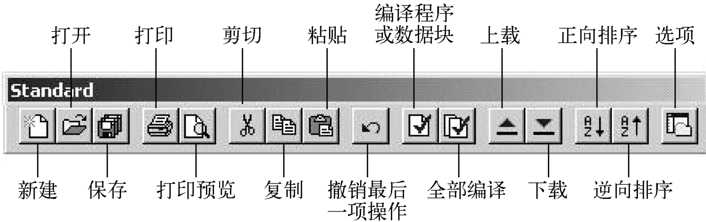


图 3-11 标准工具栏

（2）调试工具栏

调试工具栏如图 3-12 所示，在调试程序时非常有用。其中，“运行”按钮是将 PLC 设置成“运行”模式，调试时使用比较方便，也可以直接将 PLC 上的旋钮拨到“运行”模式。“停止”按钮是将 PLC 设置成“停止”模式，准备将程序下载到 PLC 之前，应将 PLC 设置成“停止”模式，也可以直接将 PLC 上的旋钮拨到“停止”模式实现。

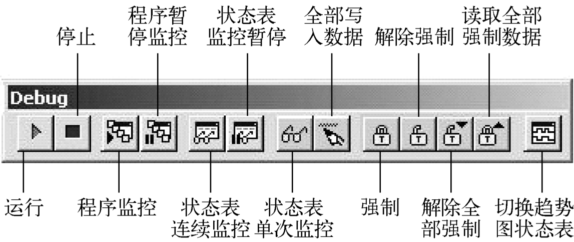


图 3-12 调试工具栏

（3）常用工具栏

常用工具栏如图 3-13 所示。其中，“插入网络”按钮最为常用，单击此按钮可以在程序中插入一个新网络。

（4）指令工具栏

指令工具栏如图 3-14 所示。在输入梯形图指令时，可以使用指令工具栏中的按钮。

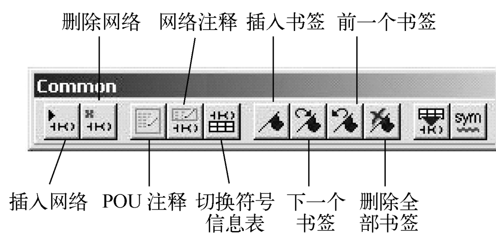


图 3-13　常用工具栏

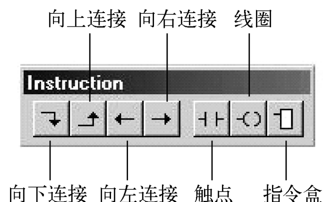


图 3-14　指令工具栏

5. 输出窗口

输出窗口保留一份最近编译的 POU 和在编译过程中产生的错误列表。如果已打开程序编辑器窗口和输出窗口，可以双击输出窗口中的错误信息，自动将程序滚动至发生错误的网络，如图 3-15 所示，在输出窗口中明确指出错误发生的位置（错误在网络 2 的行 2，列 1 处）。



图 3-15　输出窗口

修正程序后，执行新的编译，更新输出窗口，并清除已改正的网络的错误参考。

将鼠标放在输出窗口中，用鼠标右键单击，隐藏输出窗口或清除其内容。

使用“查看”（View）→“框架”（Frame）→“输出窗口”（Output Window），打开输出窗口。

6. 状态条

位于主程序底部的状态条提供有关在 STEP 7 - Micro/WIN 中操作的信息。当在编辑模式中工作时，显示编辑器信息，状态条如图 3-16 所示，光标位于网络 3 的行 1，列 1 处。状态栏根据具体情形显示下列信息：



图 3-16　状态条

① 简要状态说明。
② 当前网络号码。
③ 光标位置（用于 STL 编辑器的行和列；用于 LAD 或 FBD 编辑器的行和列）。
④ 当前编辑模式，插入或覆盖。
⑤ 表示背景任务状态的图标，例如保存或打印。

7. 键盘常用功能键

键盘常用功能键与功能对应见表 3-1。

表 3-1　键盘常用功能键与功能对应

序　　号	快捷键（键盘的功能键）	说　　明
1	Home	将光标移至同行的第一列
2	End	将光标移至同行的最后一列
3	Page Up	垂直向上移动一个屏幕
4	Page Down	垂直向下移动一个屏幕
5	左键“←”	将光标向左移动一个单元格
6	右键“→”	将光标向右移动一个单元格
7	上键“↑”	将光标向上移动一个单元格
8	下键“↓”	将光标向下移动一个单元格
9	Ctrl + Home	将光标移至第一个网络的第一个单元格
10	Ctrl + End	将光标移至最后一个网络的最后一个单元格
11	F1	当前单元格指令的帮助
12	Ctrl + Page Up	显示下一个 POU，向左移动浏览 POU 标记
13	Ctrl + Page Down	显示下一个 POU，向右移动浏览 POU 标记

3.2.3　创建新工程

新建工程有两种方法：一种方法是单击菜单栏中的“文件”→“新建”，即可新建工程，如图 3-17 所示；另一种方法是单击工具栏中的图标即可。

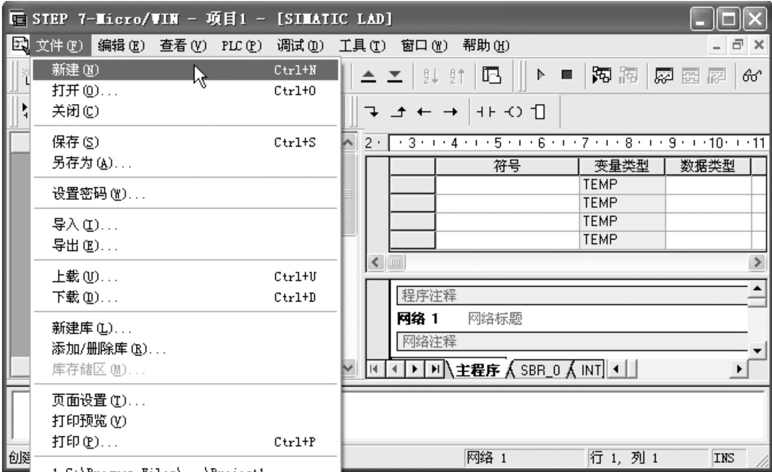


图 3-17　新建工程

3.2.4　保存工程

保存工程有两种方法：一种方法是单击菜单栏中的“文件”→“保存”，即可保存工程，如图 3-18 所示；另一种方法是单击工具栏中的图标即可。

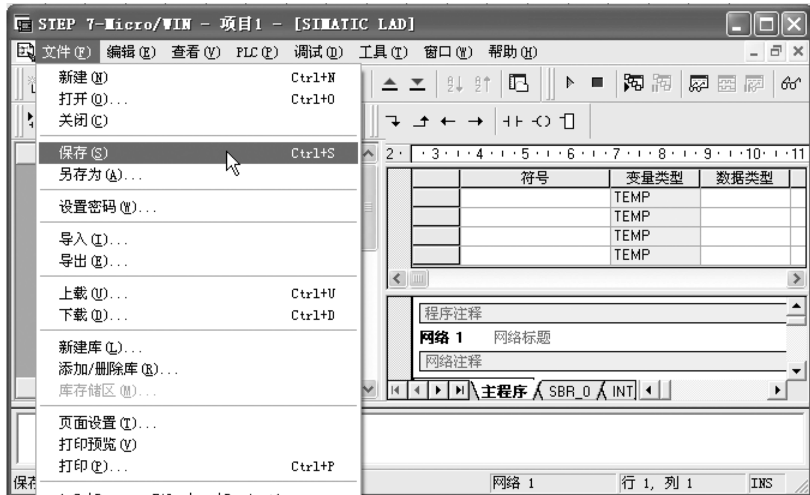


图 3-18　保存工程

3.2.5　打开工程

打开工程有 3 种方法：一种方法是单击菜单栏中的“文件”→“打开”，如图 3-19 所示，找到要打开的文件的位置，选中要打开的文件，单击“打开”按钮即可打开工程，如图 3-20 所示；第二种方法是单击工具栏中的图标，即可打开工程；第三种方法是直接在工程的存放目录下双击该工程，也可以打开此工程。

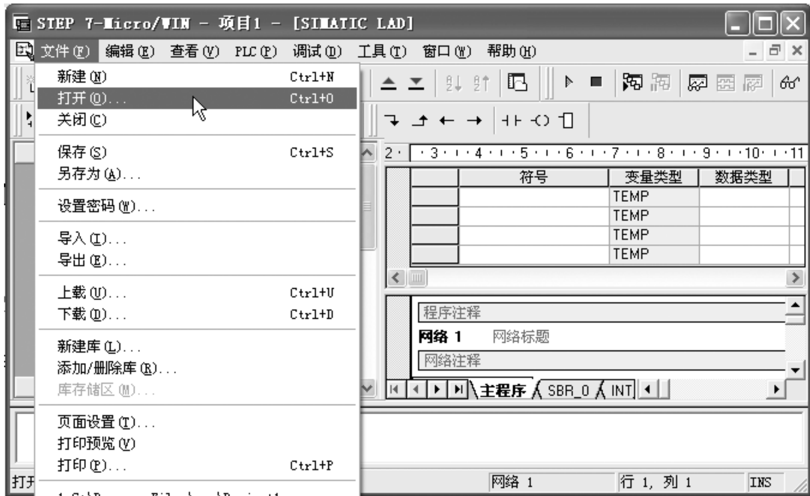


图 3-19　打开工程（1）

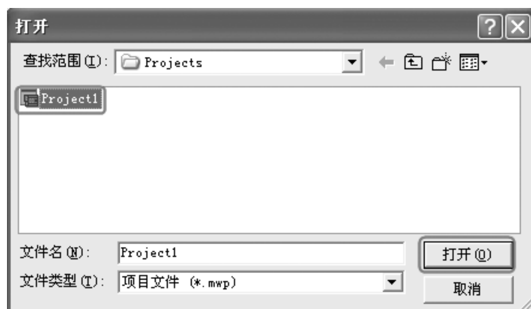


图 3-20　打开工程（2）

3.2.6　系统块的设置

S7-200 CPU 提供了多种参数和选项设置以适应具体应用，这些参数和选项在“系统块”对话框内设置。系统块必须下载到 CPU 中才起作用。有的初学者修改程序后往往不会忘记重新下载程序，而在软件中更改参数后却忘记了重新下载，这是不对的。

单击工具浏览条的“查看”视图中的“系统块”图标，或者使用菜单栏中的“查看”→“组件”→“系统块”命令，打开“系统块”对话框，如图 3-21 所示。



图 3-21　“系统块”对话框

1. 设置通信端口

在“系统块”对话框中，单击“系统块”节点下的“通信端口”，可打开“通信端口”选项卡，设置 CPU 的通信端口属性，如图 3-22 所示。

PLC 的默认地址为 2，但 PLC 通信时，通信端口的地址不能重复，通信端口的地址必须是唯一的（同一台 PLC 的两个端口的地址一般相同），因此需要更改 PLC 的地址。波特率必须和开始设置的传输率一致。更改完成后，必须下载到 CPU 中才起作用。当然，使用指令“SET_ADDR”也可以更改通信端口的地址，但必须运行程序。

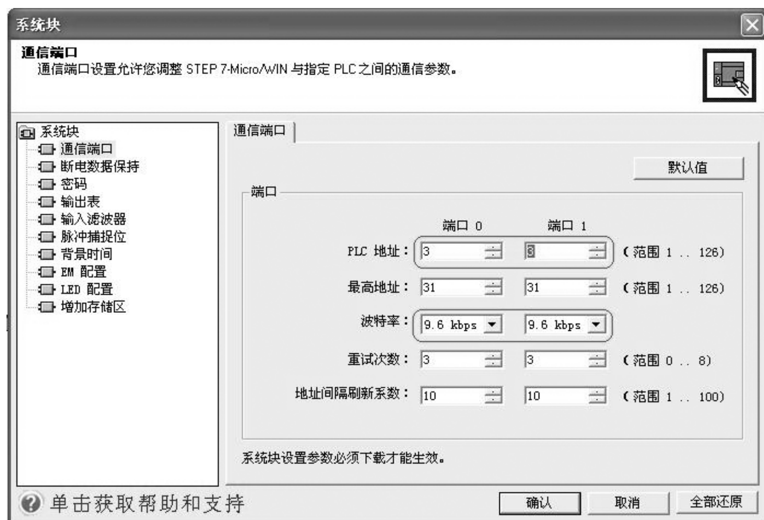


图 3-22　设置通信端口

2. 设置断电数据保持

在“系统块”对话框中，单击“系统块”节点下的“断电数据保持”，可打开“断电数据保持”对话框，如图 3-23 所示。断电数据保持设置就是定义 CPU 如何处理各数据区的数据保持任务。在数据保持设置区中选中的就是要保持其数据内容的数据区。所谓“保持”就是在 CPU 断电后再上电，数据区域的内容是否保持断电前的状态。在这里设置的数据保持功能依靠如下几种方式实现。



图 3-23　设置断电数据保持

① CPU 的内置超级电容，在断电时间不太长时，可以为数据和时钟的保持提供电源缓冲。

② CPU 上可以附加电池卡，与内置电容配合，长期为时钟和数据保持提供电源。

③ 设置系统块，在 CPU 断电时自动保存 M 区中 14 B 的数据。

④ 在数据块中定义不需要更改的数据，下载到 CPU 内可以永久保存。

⑤ 用户编程使用相应的特殊寄存器功能，将数据写入 EEPROM 永久保存。

如果将 MB0 ~ MB13 共 14 B 范围中的存储单元设置为“保持”，则 CPU 在断电时会自动将其内容写入 EEPROM 的相应区域中，在重新上电后用 EEPROM 的内容覆盖这些存储区。如果将其他数据区的范围设置为“不保持”，CPU 会在重新上电后将 EEPROM 中的数值复制到相应的地址；如果将数据区的范围设置为“保持”，一旦内置超级电容（+电池卡）未能成功保持数据，则会将 EEPROM 的内容覆盖相应的数据区，反之则不覆盖。

如果关断 CPU 的电源再上电，观察到 V 存储区的相应的单元内还保存有正确的数据，则可说明数据已经成功地写入 CPU 的 EEPROM。

3. 设置密码

通过设置密码可以限制对 S7-200 CPU 的内容的访问。在“系统块”对话框中，单击“系统块”节点下的“密码”，可打开“密码”选项卡，设置密码保护功能，如图 3-24 所示。密码的保护等级分为 4 个等级，除了“全部权限（1 级）”外，其他的均需要在“密码”和“验证”文本框中输入起保护作用的密码。

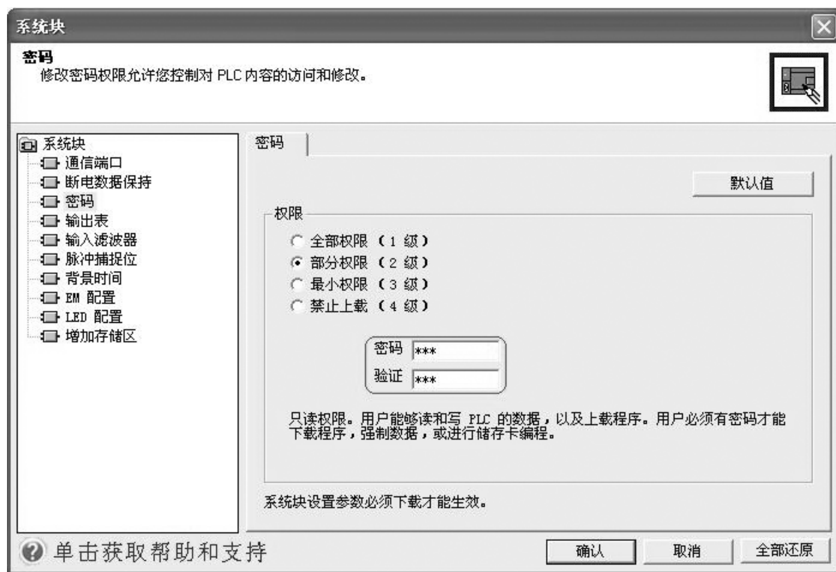


图 3-24 设置密码

要检验密码是否生效，可以进行以下操作。

① 停止 Micro/WIN 与 CPU 的通信 1 min 以上。

② 关闭 Micro/WIN 程序，再打开。

③ 停止 CPU 的供电，再送电。

如果忘记了密码，必须清除 CPU 的内存才能重新下载程序。执行清除 CPU 指令并不会改变 CPU 原有的网络地址、波特率和实时时钟；如果有外插程序存储卡，其内容也不会改变。清除密码后，CPU 中原有的程序将不存在。要清除密码，可按如下 3 种方法操作。

① 在 Micro/WIN 中选择“PLC”→“Clear”，选择程序块、数据块和系统块，并按“OK”按钮确认。

② 另外一种方法是通过程序 wipeout. exe 来恢复 CPU 的默认设置。这个程序可在 STEP 7 - Micro/WIN 安装光盘中找到。

③ 此外，还可以在 CPU 上插入一个含有未加密程序的外插存储卡，上电后此程序会自动装入 CPU 并且覆盖原有的带密码的程序，然后 CPU 可以自由访问。

西门子公司随编程软件 Micro/WIN 提供的库指令、指令向导生成的子程序、中断程序都进行了加密。加密并不妨碍使用它们。加密的程序会显示一个锁形标记，不能打开查看程序内容。将加密的程序下载到 CPU 中，再上传后也保持加密状态。

如果用户想保护编写的程序项目，可以使用“文件”→“设置密码”命令来保存程序项目。

【关键点】 PLC 的软件加密比较容易被破解，不能绝对保证程序的安全，目前网络上有一些破解软件可以轻易破解 PLC 的用户程序的密码，编者强烈建议读者在保护自身权益的同时，必须尊重他人的知识产权。

3. 2. 7　数据块

数据块用于为 V 存储器指定初始值。可使用不同的长度（字节、字或双字）在 V 存储器中保存不同格式的数据。单击工具浏览条的“查看”视图中的“数据块”图标，或者单击菜单栏中的“查看”→“组件”→“数据块”命令打开“数据块”窗口。在图 3-25 中输入“VB0 100”和“VW2 100”两行数据，实际上就是起初始化的作用，与图 3-26 中的梯形图程序的作用相同。



图 3-25　“数据块”窗口

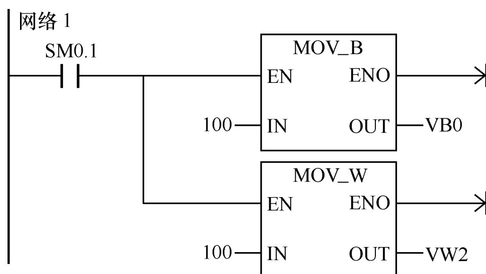


图 3-26　初始化程序

数据块必须下载到 CPU 中才起作用，数据块保存在 CPU 的 EEPROM 存储单元中，因此断电后仍然能保持数据。

3.2.8 程序调试

程序调试是工程中的一个重要步骤，因为初步编写完成的程序不一定正确，有时虽然逻辑正确，但需要修改参数，因此程序调试十分重要。Micro/WIN 提供了丰富的程序调试工具供用户使用，下面分别进行介绍。

1. 状态表

使用状态表可以监控数据，各种参数（如 CPU 的 I/O 开关状态、模拟量的当前数值等）都在状态表中显示。此外，配合“强制”功能还能将相关数据写入 CPU，改变参数的状态，例如可以改变 I/O 开关状态。

单击工具浏览条的“查看”视图中的“状态表”图标，弹出“状态表”窗口，单击菜单栏中的“查看”→“组件”→“状态表”命令也可以打开，如图 3-27 所示。在其中可以设置相关参数，单击工具栏中的“状态表监控”按钮可以监控数据。

状态表

	地址	格式	当前值	新值
1	I0.0	位	2#0	
2	I0.1	位	2#0	
3	Q0.0	位	2#0	
4		有符号		
5		有符号		

图 3-27 “状态表”窗口

2. 强制

S7-200 系列 PLC 提供了强制功能，以方便调试工作。在现场不具备某些外部条件的情况下模拟工艺状态。用户可以对数字量（DI/DO）和模拟量（AI/AO）进行强制。强制时，运行状态指示灯变成黄色，取消强制后指示灯变成绿色。

如果在没有实际的 I/O 连线时，可以利用强制功能调试程序。先打开“状态表”窗口并使其处于监控状态，在“新值”数值框中写入要强制的数据，然后单击工具栏中的“强制”按钮，此时，被强制的变量数值上有一个标志，如图 3-28 所示。

状态表

	地址	格式	当前值	新值
1	I0.0	位	2#1	
2	I0.1	位	2#0	
3	Q0.0	位	2#1	
4		有符号		
5		有符号		

图 3-28 使用强制功能

单击工具栏中的“取消全部强制”按钮可以取消全部的强制。

3. 写入数据

S7-200 系列 PLC 提供了数据写入功能，以方便调试工作。例如，在“状态表”窗口中输入 Q0.0 的新值“0”，如图 3-29 所示，单击工具栏上的“全部写入”按钮，或者单

击菜单栏中的“调试”→“全部写入”命令即可更新数据。

状态表

	地址	格式	当前值	新值
1	I0.0	位	2#0	
2	I0.1	位	2#0	
3	Q0.0	位	2#1	0
4		有符号		
5		有符号		

图 3-29　写入数据

利用“全部写入”功能可以同时输入几个数据。“全部写入”的作用类似于“强制”的作用。但两者是有区别的：强制功能的优先级别要高于“全部写入”，“全部写入”的数据可能改变参数状态，但当与逻辑运算的结果抵触时，写入的数值也可能不起作用。

【例 3-1】 如图 3-30 所示的梯形图，Q0.0 状态为 1，问在“状态表”中，分别用“写入”、“强制”功能，是否能将 Q0.0 的数值变成 0？

【解】

用“写入”功能不能将 Q0.0 的数值变成 0，因为图 3-30 的梯形图的逻辑运算的结果造成 Q0.0 为 1，与“修改”结果抵触，最后输出结果以逻辑运算的结果为准。

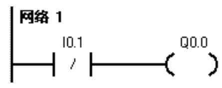


图 3-30　梯形图

用“强制”功能将 Q0.0 的数值变成 0，因为强制的作用高于逻辑运算的作用。

4. 趋势图

前面提到的状态表可以监控数据，趋势图同样可以监控数据，只不过使用状态表监控数据时的结果是以表格的形式表示的，而使用趋势图时则以曲线的形式表达。利用后者能够更加直观地观察数字量信号变化的逻辑时序或者模拟量的变化趋势。

单击调试工具栏上的“切换趋势图状态表”按钮可以在状态表和趋势图形式之间切换，趋势图如图 3-31 所示。

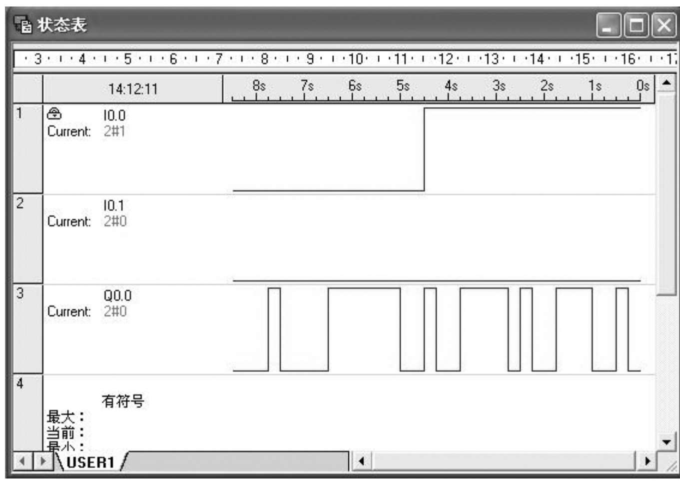


图 3-31　趋势图

趋势图对变量的反应速度取决于 Micro/WIN 与 CPU 通信的速度以及图中的时间基准。在趋势图中单击可以选择图形更新的速率。当停止监控时，可以冻结图形以便仔细分析。

3.2.9 交叉引用

交叉引用表能显示程序中元件使用的详细信息。交叉引用表对查找程序中数据地址的使用十分有用。在工具浏览条的“查看”视图下单击“交叉引用”图标，可弹出如图 3-32 所示的界面。当双击交叉引用表中某个元素时，界面立即切换到程序编辑器中显示交叉引用对应元件的程序段。例如，双击“交叉引用表”中第一行的“I0.0”，界面切换到程序编辑器中，而且光标（方框）停留在“I0.0”上，如图 3-33 所示。

	元素	块	位置	关联
1	I0.0	程序块 (OB1)	网络 1	-| |-
2	I0.0	程序块 (OB1)	网络 2	-| |-
3	Q0.0	程序块 (OB1)	网络 1	-( )
4	VB10	程序块 (OB1)	网络 2	MOV_B

图 3-32　交叉引用表

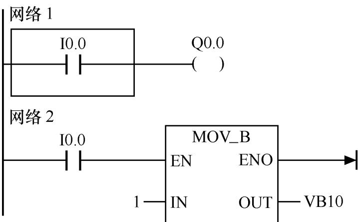


图 3-33　交叉引用表对应的程序

3.2.10 符号表

符号是可为存储器地址或常量指定的符号名称。符号表是符号和地址对应关系的列表。打开符号表有三种方法，具体如下：

① 单击浏览条中的“符号表”按钮。

② 选择“查看”（View）→“符号表”（Symbol Table）菜单命令。

③ 打开指令树中的符号表或全局变量文件夹，然后双击一个表格图标。

可以使用项目中的系统符号表。预定义的系统符号表提供了常用 PLC 系统功能访问的快捷参考。PLC 系统符号将功能名称与用于激活该功能的 PLC 特殊存储区位置相联系。

【例 3-2】 如图 3-34 所示，是一段简单的程序，要求显示其符号信息表和符号表，请写出操作过程。

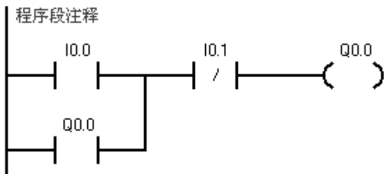


图 3-34　程序

首先，在项目树中展开“符号表”，双击“用户定义 1”弹出符号表，如图 3-35 所示，在符号表中，按照如图 3-35 所示填写。符号“起动”实际就代表地址“I0. 0”，符号“停止”实际就代表地址“I0. 1”，符号电动机”实际就代表地址“Q0. 0”。



图 3-35　打开符号表

接着，在视图功能区，单击“查看”→“符号表”→“将符号应用于项目”按钮。此时，符号和地址的对应关系显示在梯形图中，如图 3-36 所示。

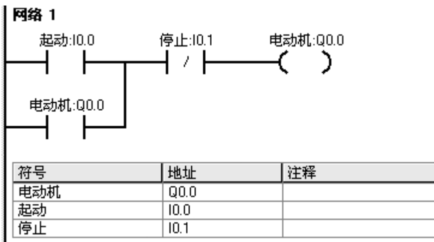


图 3-36　信息符号表

3.2.11　变量表

初学者一般不会用到变量表，以下用一个例子来说明变量表的使用。

【例 3-3】 用子程序表达算式 Ly = (La - Lb) × Lx。

【解】

1）首先打开子程序“SBR_0”，子程序的上方就是变量表。

2）在变量表中，输入如图 3-37 所示的参数。

	符号	变量类型	数据类型	注释
	EN	IN	BOOL	
LW0	La	IN	INT	
LW2	Lb	IN	INT	
LW4	Lx	IN	INT	
		IN		
		IN_OUT		
LD6	Ly	OUT	DINT	
		OUT		

图 3-37　变量表

3）在子程序中输入如图 3-38 所示的程序。

4）在主程序中调用子程序，并将运算结果存入 VD0 中，如图 3-39 所示。

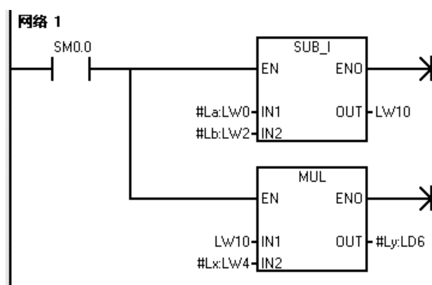


图 3-38　子程序

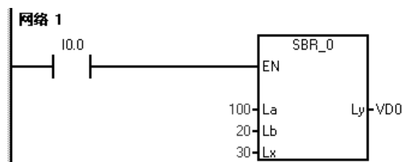


图 3-39　主程序

【关键点】 LW0、LW2、LW4 的数据类型是"INT"，变量类型是"输入"（IN）；而 LD6 的数据类型是"DINT"，变量类型是"输出"（OUT），不能弄错。

这个例子读者可以在学完后续课程后，再学习。

3.2.12　工具浏览条

STEP 7 – Micro/WIN 的工具浏览条中有指令向导、文本显示向导、位置控制向导、PID 控制面板、以太网向导和 EM 253 控制面板等工具。这些工具很实用，有的工具能使比较复杂的编程变得简单，例如，使用"指令向导"工具的网络读写指令向导，就能将较复杂的网络读写指令通过向导指引生成子程序；有的工具的功能则是不能取代的，例如，使用以太网模块进行通信时就必须使用"以太网向导"工具。

3.2.13　帮助菜单

STEP 7 – Micro/WIN 软件虽然界面友好，比较容易使用，但遇到问题是难免的。STEP 7 – Micro/WIN 软件提供了详尽的帮助。使用菜单栏中的"帮助"→"目录和索引"命令可以打开如图 3-40 所示的"帮助"对话框。其中有两个选项卡，分别是"目录"和"索引"。"目录"选项卡中显示的是 STEP 7 – Micro/WIN 软件的帮助主题，单击帮助主题可以查看详细内容；"索引"选项卡中，可以根据关键字查询帮助主题。

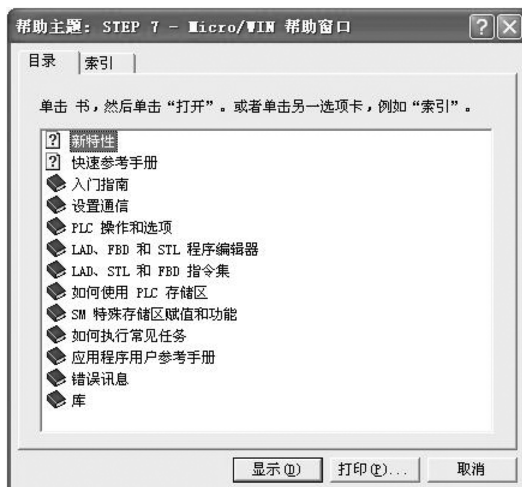


图 3-40　使用 STEP 7 – Micro/WIN 的帮助

3.3 用户自定义指令库

读者可以使用西门子公司编写的库，典型的是 USS 指令库和 Modbus 指令库，也可以自己定义指令库。用户可以把自己编制程序集成到编程软件 STEP 7 - Micro/WIN 中。这样可以在编程时调用实现相同功能的库指令，而不必同时打开几个项目文件拷贝。指令库也可以方便地在多个编程计算机之间传递。

一个已存在的程序项目只有子程序、中断程序可以被创建为指令库。中断程序只能随定义它的主程序、子程序集成到库中。指令库的使用方法与子程序基本一样。

【例 3-4】将例 3-3 中的子程序定义成库函数。

【解】

定义指令库和调用指令的具体方法如下：

1. 创建库

在菜单栏中，单击“文件”→“新建库”，如图 3-41 所示，打开“创建库”界面，如图 3-42 所示，在“组件”选项卡中，选择已经创建好的程序，本例为“函数 1”，单击“添加”按钮。选中“属性”选项卡，如图 3-43 所示，输入库名“函数 1”，再单击“浏览”按钮，弹出如图 3-44 所示的界面，单击“保存”按钮，保存库文件。回到图 3-42 所示的界面，单击“确定”按钮，完成创建库。



图 3-41 “创建库”界面（1）

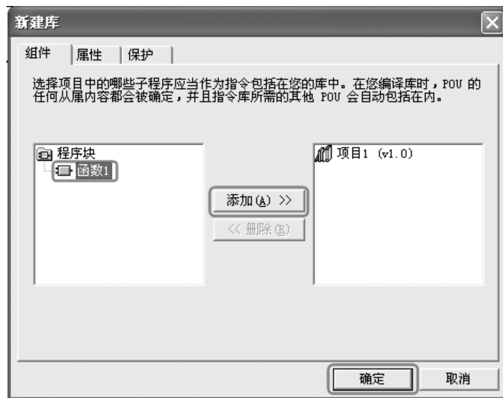


图 3-42 “创建库”界面（2）

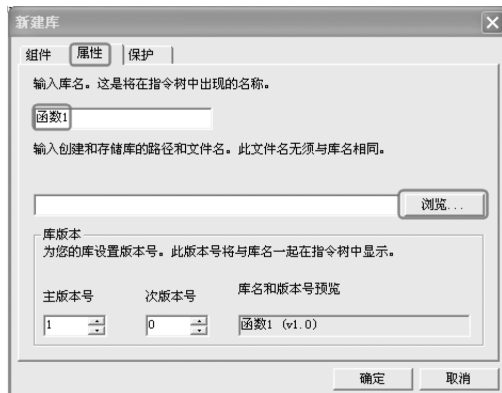


图 3-43 “创建库”界面（3）

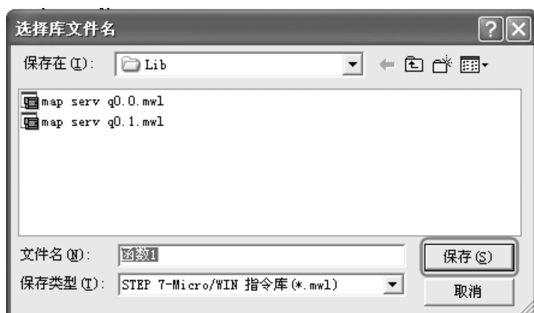


图 3-44　保存“库文件”

2. 添加库文件

在菜单栏中，单击“文件”→“添加/删除库”，如图 3-45 所示，将弹出如图 3-46 所示界面，单击“添加”按钮，弹出如图 3-47 所示界面，选择“函数 1”文件，单击“确认”按钮，此时“函数 1”已经添加到“库”中。

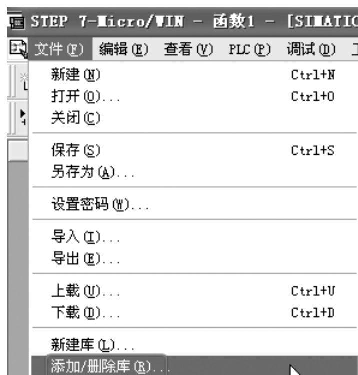


图 3-45　添加库文件（1）

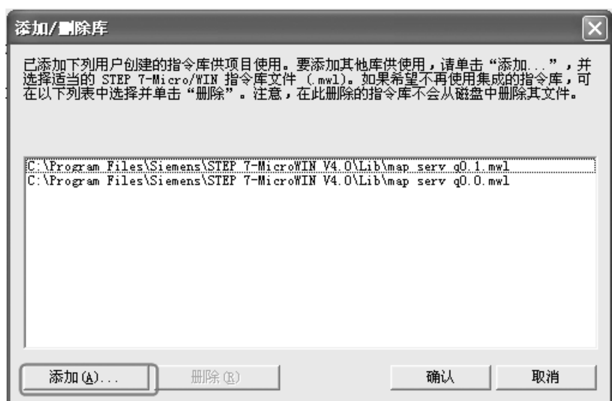


图 3-46　添加库文件（2）

3. 查看“库文件”

在“项目树”中，展开“库”，可以看到“函数 1”已经添加到“库”中，如图 3-48 所示。

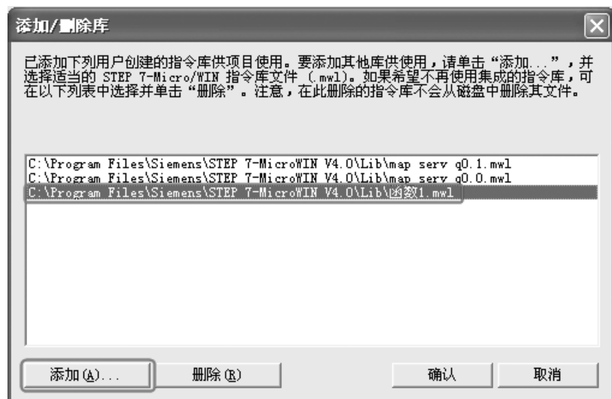


图 3-47　添加库文件（3）

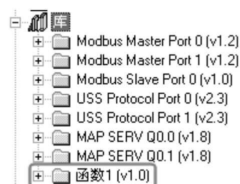


图 3-48　查看“库文件”

【关键点】

① 一个已存在的程序项目，只有子程序、中断程序可以被创建为指令库，主程序不能被创建成库。

② 如果不小心将存放的库函数删除，则以后不能被成功调用。

3.4　S7－200 PLC 扩展模块的地址分配

在编写程序的时候，正确分配 S7－200 PLC 的扩展模的地址是至关重要的，否则不能编写出正确的程序。

3.4.1　模块的地址分配

1）在扩展时，扩展模块的位置可以任意放置。

2）分配地址时，各模块的位置按照各自的类型进行排队，数字量输入和数字量输出互不影响，数字量和模拟量输出互不影响。

3）数字量模块以 8 点，即一个字节单位分配给模块，不足一个字节将空置。如图 3-49 所示，CPU 224XP 和一个 4 入/4 出的混合数字量模块组成一个系统。CPU 224XP 的输入是 14 点，占用 2 个字节（即 16 点），而扩展模块，是 4 点输入，占用 1 个字节（即 8 点）。CPU 224XP 的 I1.6 和 I1.7 空置不用，扩展的 I2.4～I2.7 也空置不用。

如图 3-50 所示，CPU 224XP 和一个 4 入/4 出的混合数字量模块组成一个系统。CPU 224XP 的输出是 10 点，占用 2 个字节（即 16 点），而扩展模块，是 4 点输出，占用 1 个字节（即 8 点）。CPU 224XP 的 Q1.0～Q1.7 空置不用，扩展的 Q2.4～Q2.7 也空置不用。

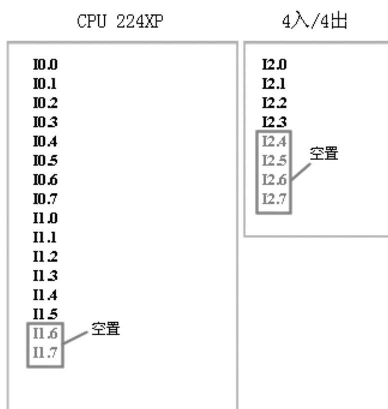


图 3-49　数字量输入地址分配

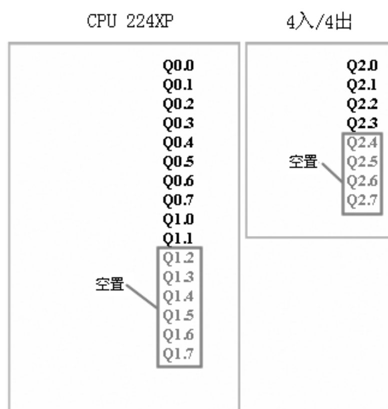


图 3-50　数字量输出地址分配

4）模拟量模块以 2 点，即 1 个字为单位分配地址给模块。如图 3-51 所示，CPU 224XP 和一个 4 入/1 出的混合模拟量模块组成一个系统。CPU 224XP 的输入是 4 点，占用 2 个字，而扩展模块，是 4 点输入，占用 4 个字。注意：模拟量是以 1 个字（2 个字节）向上增加的，即 AIW0、AIW2、AIW4，且为偶数，不能是 AIW1、AIW3、AIW5。模拟量输出也是类似的。

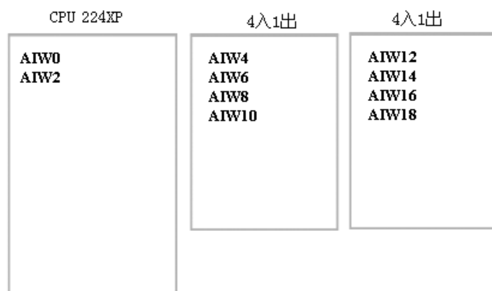


图 3-51　模拟量输入地址分配

3.4.2　模块的地址查询

在上面的章节中，对模块地址的分配机理进行了讲述，如果读者还不能确定自己的地址分配是否正确，那么读者可以通过查询，查看软件自动分配的地址。

要查询地址，首先要将模块安装到位，并让 CPU 和 STEP 7 - Micro/WIN 4.0 进行通信。在菜单中，单击“PLC”→“信息”，如图 3-52 所示。弹出“PLC 信息”对话框，如图 3-53 所示。

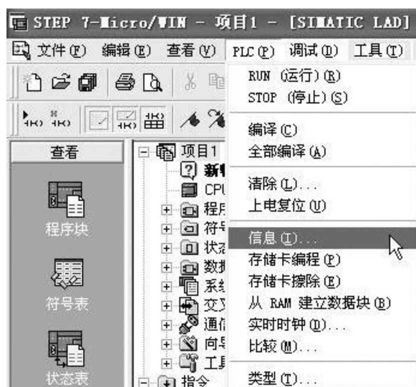


图 3-52　打开“PLC 信息”对话框

在图 3-53 中，显示的是：PLC 模块的数字量输入是 24 点，起始地址是 I0.0，数字量输出是 16 点，起始地址是 Q0.0；扩展模块 0 是模拟量模块，4 个点的模拟量输入，起始地址是 AIW0，无模拟量输出；扩展模块 1 是模拟量模块，2 个点的模拟量输出，起始地址是 AQW0，无模拟量输入；扩展模块 2 是 EM 277 模块，不占用任何地址。

【例 3-5】 某系统由 4 个模块组成，其安装顺序是 CPU 226、EM 223（4 入/4 出）、EM 235 和 EM 231，则分配地址分别是多少？

【解】

各模块的地址如下：

1）CPU 226：数字量输入地址是 I0.0 ~ I0.7；I1.0 ~ I1.7；I2.0 ~ I2.7。数字量输出地址是 Q0.0 ~ I0.7；Q1.0 ~ I1.7。

2）EM 223（4 入/4 出）：数字量输入地址是 I3.0 ~ I3.3。数字量输出地址是 Q2.0 ~ I2.3。



图 3-53　PLC 信息

3）EM 235：模拟量输入地址是 AIW0、AIW2、AIW4 和 AIW6。模拟量输出地址是 AQW0。

4）EM 231：模拟量输入地址是 AIW8、AIW10、AIW12 和 AIW14。

3.5　用 STEP 7 – Micro/WIN 建立一个完整的项目

下面以图 3-54 所示的启/停控制梯形图为例，完整地介绍一个程序从输入到下载、运行和监控的全过程，说明 STEP 7 – Micro/WIN 软件的使用方法。

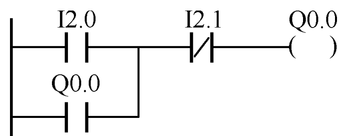


图 3-54　启/停控制梯形图

1. 启动 STEP 7 – Micro/WIN 软件

启动 STEP 7 – Micro/WIN 软件，弹出如图 3-55 所示界面。

2. 切换成中文界面

很多用户更喜欢中文界面，STEP 7 – Micro/WIN 软件提供了德语、英语、汉语等 6 种语言供用户选择。单击菜单栏中的“Tools”→“Options”命令，弹出如图 3-56 所示的对话框。选中“Options”节点下的“General”，在右侧的“Language”列表框中选中所需要的语

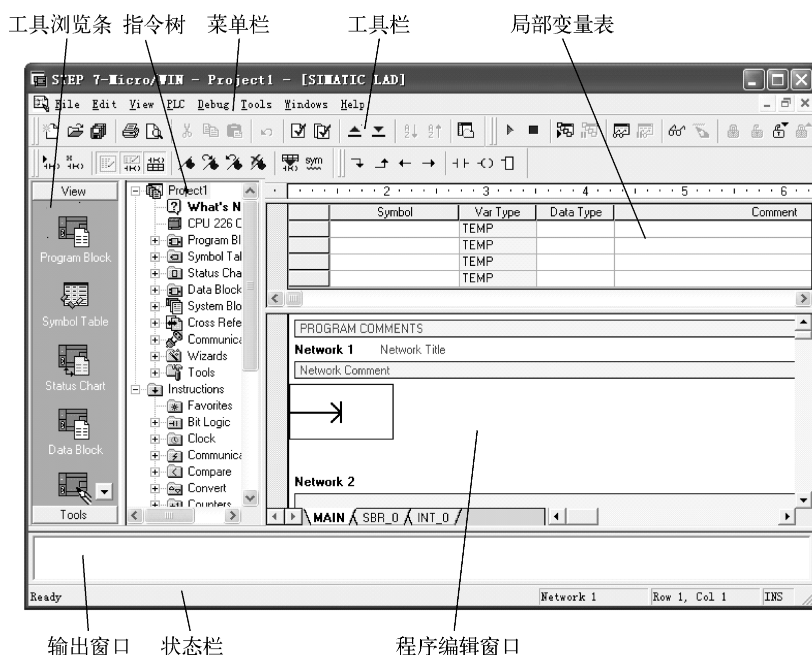


图 3-55　STEP 7 - Micro/WIN 软件初始界面

言“Chinese”，单击“OK”按钮。这时弹出如图 3-57 所示的对话框，单击“确定”按钮，接着弹出如图 3-58 所示的对话框，单击“是”按钮，软件自动关闭。下一次运行 STEP 7 - Micro/WIN 软件时，将自动出现中文界面。

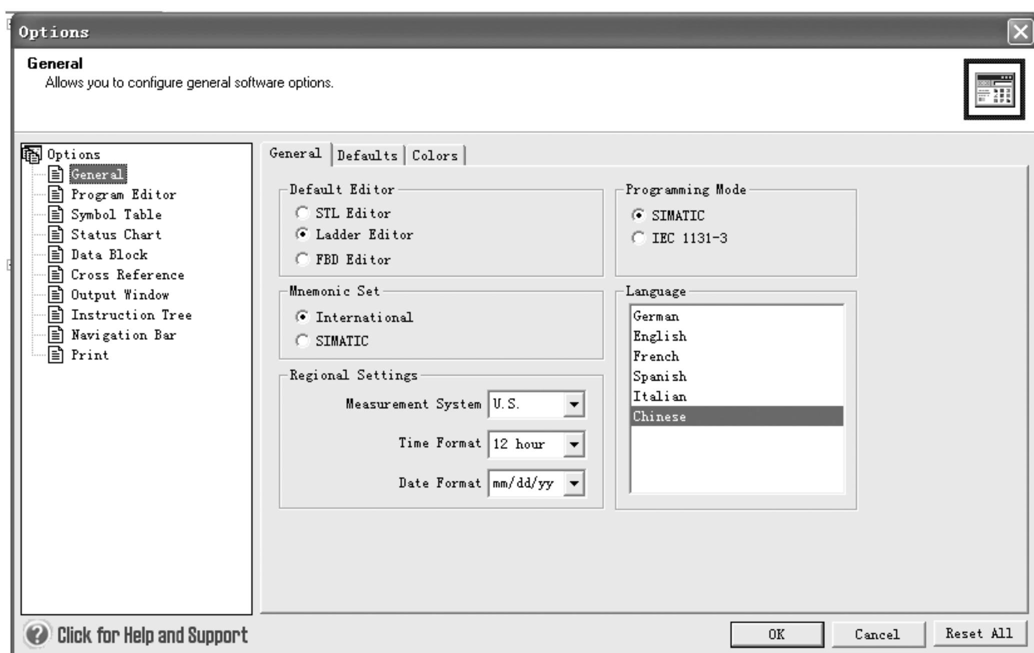


图 3-56　设置所需要的语言



图 3-57　确认改变选项界面



图 3-58　保存项目界面

3. PLC 的类型选择

展开指令树中的“项目 1”节点，选中并双击“CPU 2XX”（可能是 CPU 221），这时弹出“PLC 类型”对话框，在“PLC 类型”下拉列表框中选定“CPU 226 CN”（这是本例的机型），然后单击“确认”按钮，如图 3-59 所示。

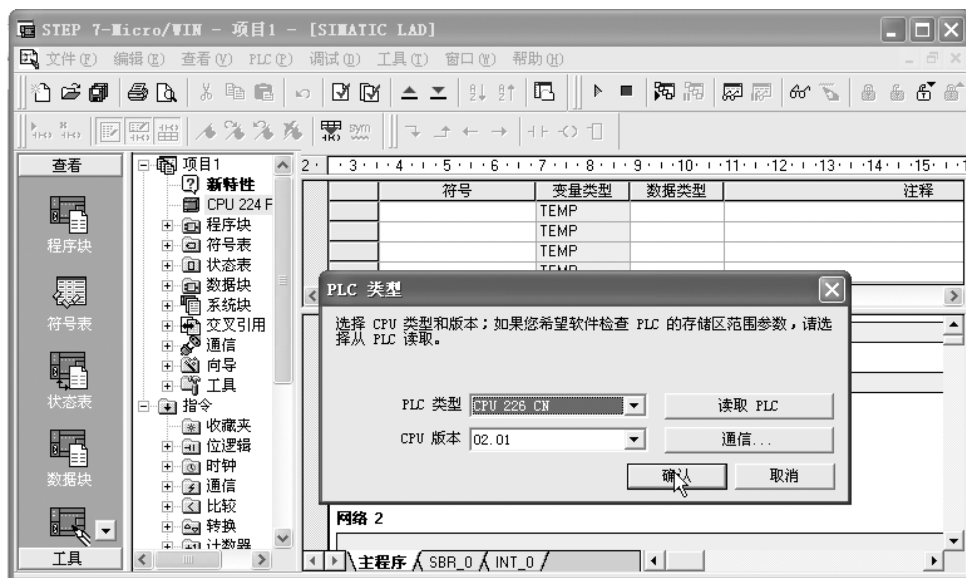


图 3-59　PLC 类型选择界面

4. 输入程序

展开指令树中的“指令”节点，依次双击常开触点按钮“┤├”（或者拖入程序编辑窗口）、常闭触点按钮“┤/├”、输出线圈按钮“（ ）”，换行后再双击常开触点按钮“┤├”，出现程序输入界面，如图 3-60 所示。接着单击红色的问号，输入寄存器及其地址（本例为 I2.0、Q0.0 等），输入完毕后如图 3-61 所示。

【关键点】 有的初学者在输入时会犯这样的错误，将“Q0.0”错误地输入成“QO.o”，此时“QO.o”下面将有红色的波浪线提示错误。

5. 编译程序

单击标准工具栏的“全部编译”按钮☑进行编译，若程序有错误，则输出窗口会显示错误信息。

编译后如果有错误，可在下方的输出窗口查看错误，双击该错误即跳转到程序中该错误的所在处，根据系统手册中的指令要求进行修改，如图 3-62 所示。

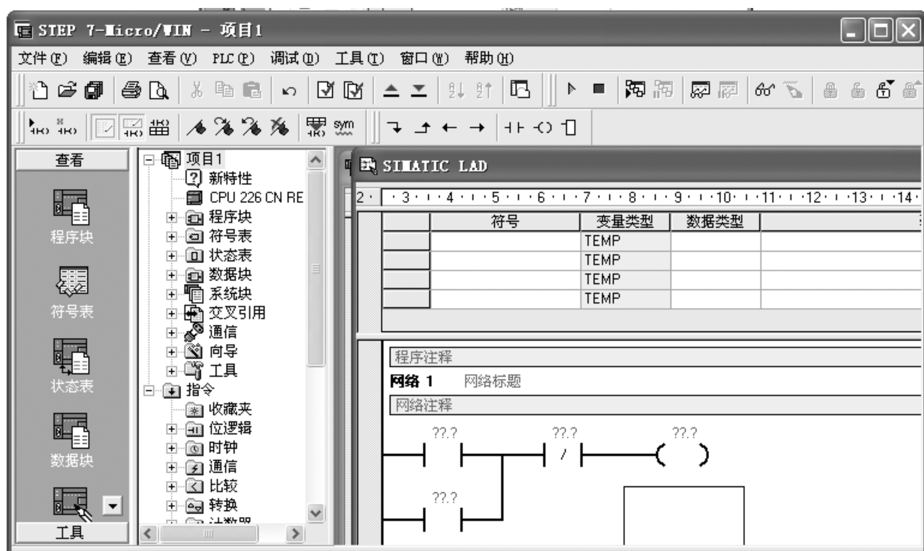


图 3-60　程序输入界面

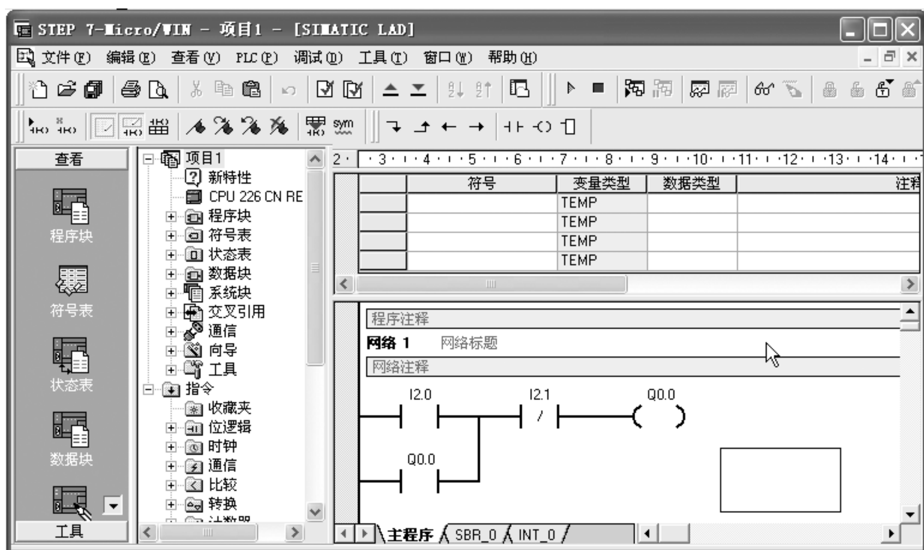


图 3-61　输入程序界面

6. 设置通信

单击工具浏览条中“查看”视图中的“设置 PG/PC 接口”图标，弹出“设置 PG/PC 接口”对话框，在“为使用的接口分配参数”列表框中选择“PC/PPI cable（PPI）”选项并双击，弹出“属性 - PC/PPI cable（PPI）”对话框，可使用默认数值，如图 3-63 所示。接着选择“本地连接”选项卡，在“连接到”下拉列表框中选择编程电缆与计算机相连的接口，本例为“COM1”，再单击“确定”按钮，如图 3-64 所示。注意：传输率一定要与通信电缆上的设置一致，否则不能建立通信。

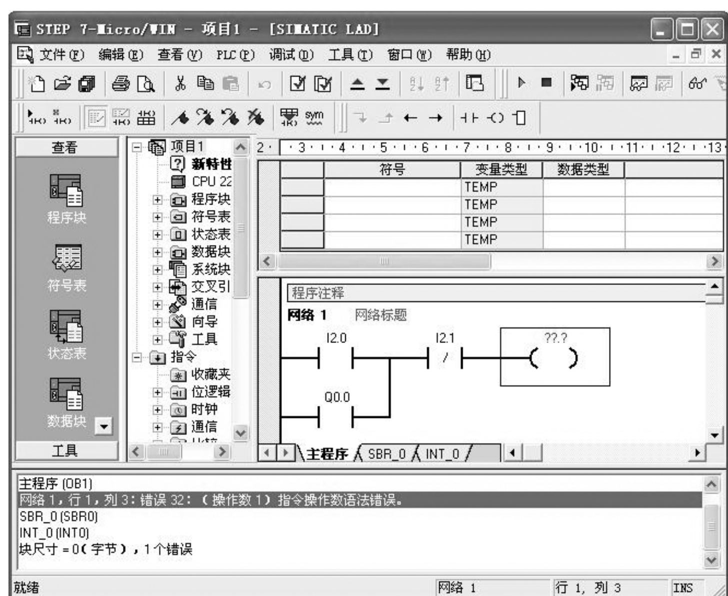


图 3-62　编译程序



图 3-63　设置通信参数



图 3-64　选择连接接口

初学者往往容易碰到 Micro/WIN 与 CPU 通信失败的情况，可能的原因如下：

① Micro/WIN 中设置的对方通信口地址与 CPU 的实际口地址不同。

② Micro/WIN 中设置的本地地址与 CPU 通信口的地址相同（一般应当将 Micro/WIN 的本地地址设置为“0”）。

③ Micro/WIN 使用的通信波特率与 CPU 端口的实际通信速率设置不同。

④ 有些程序会将 CPU 上的通信口设置为自由口模式，此时不能进行编程通信。编程通信是 PPI 模式，而在“STOP”状态下，通信口永远是 PPI 从站模式，因此最好把 CPU 上的模式开关拨到“STOP”的位置。

⑤ 编程电缆有问题，此时可更换一根西门子的原装 PPI 编程电缆。

⑥ 编程口烧毁，必须送修。

⑦ 英文界面时，不可以将程序下载到中国版的 CPU 中，简单说就是 CPU 后有“CN”标记，如 CPU 226CN。

有的用户用 CP 卡进行编程通信，尽管 CP 卡的功能强大，但必须注意如下问题。

① CP 5613 不能连接 S7-200 CPU 通信口编程。

② CP 5511/5512/5611 不能在 Windows XP Home 版本下使用。

③ 所有的 CP 卡都不支持 S7-200 的自由口编程调试。

④ CP 卡与 S7-200 通信时，不能选择“CP 卡（auto）”选项。

7. 连机通信

选中工具浏览条中“查看”视图下的“通信”图标并单击，弹出“通信”对话框；再双击“双击刷新”，计算机自动搜索 PLC，若找到，则自动将目标 PLC 的地址和型号等信息显示出来，如图 3-65 所示。搜索完成后，单击“确认”按钮，这时计算机与 PLC 已经可以通信了。有时搜索结果有误，原因在于远程地址和 PLC 地址不一致造成，例如本例中的远程地址和搜索的地址都为“2”。



图 3-65　连机通信

8. 下载程序

单击标准工具栏中的下载按钮 ，弹出“下载”对话框，如图 3-66 所示，单击“下载”按钮，若 PLC 此时处于“运行”模式，系统将提示用户将“选项”栏中的“程序块”、“数据块”和“系统块”3 个选项全部勾选，再将 PLC 设置成“停止”模式，然后单击“确定”按钮，则程序自动下载到 PLC 中。下载成功后，输出窗口中有“下载成功”字样的提示。



图 3-66　下载程序

9. 程序状态监控

在调试程序时，“程序状态监控”功能非常有用，当开启此功能时，闭合的触点中有蓝色的矩形，而断开的触点中没有蓝色的矩形，如图 3-67 所示。要开启“程序状态监控”功能，只需要单击调试工具栏上的“程序监控”按钮即可。

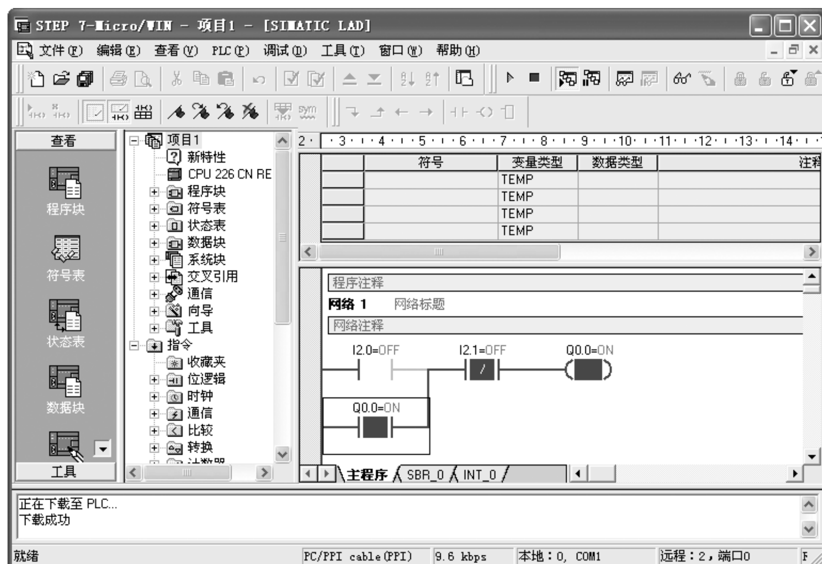


图 3-67　程序状态监控

3.6 仿真软件的使用

3.6.1 仿真软件简介

仿真软件可以在计算机或者编程设备（如 Power PG）中模拟 PLC 运行和测试程序，就像运行在真实的硬件上一样。西门子公司为 S7 - 300/400 系列 PLC 设计了仿真软件 PLCSIM，但遗憾的是没有为 S7 - 200 系列 PLC 设计仿真软件。下面将介绍应用较广泛的仿真软件 S7 - 200 SIM 2.0。

3.6.2 仿真软件 S7 - 200 SIM 2.0 的使用

S7 - 200 SIM 2.0 仿真软件的界面友好，使用非常简单，下面以如图 3-68 所示的程序的仿真为例介绍 S7 - 200 SIM 2.0 的使用步骤如下：

① 在 Micro/WIN 软件中编译如图 3 - 68 所示的程序，再单击菜单栏中的“文件”→“导出”命令，并将导出的文件保存，文件的扩展名为默认的“. awl”（文件的全名保存为 123. awl）。

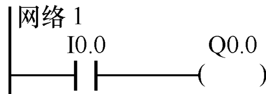


图 3-68 示例程序

② 打开 S7 - 200 SIM 2.0 软件，单击菜单栏中的“配置”→“CPU 型号”命令，弹出“CPU Type”（CPU 型号）对话框，选定所需的 CPU，如图 3-69 所示，再单击“Accept”（确定）按钮即可。

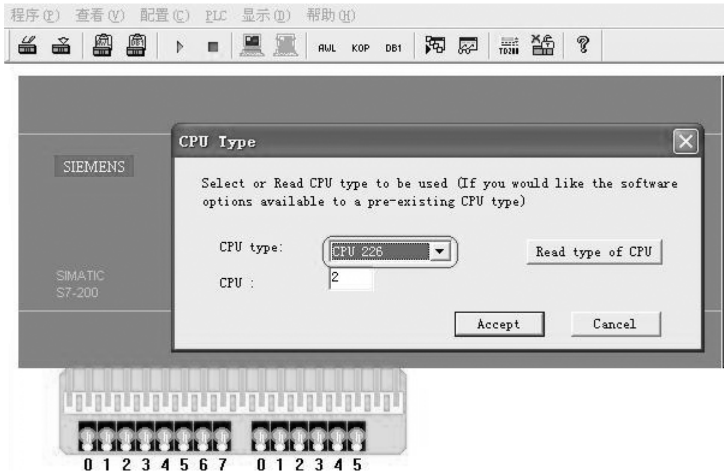


图 3-69 CPU 型号设定

③ 装载程序。单击菜单栏中的“程序”→“装载程序”命令，弹出“装载程序”对话框，设置如图 3-70 所示，再单击“确定”按钮，弹出“打开”对话框，如图 3-71 所示，

选中要装载的程序“123. awl”，最后单击“打开”按钮即可。此时，程序已经装载完成。



图 3-70　装载程序



图 3-71　打开文件

④ 开始仿真。单击工具栏上的“运行”按钮，运行指示灯亮，如图 3-72 所示，单击按钮“I0. 0”，按钮向上合上，PLC 的输入点“I0. 0”有输入，输入指示灯亮，同时输出点“Q0. 0”输出，输出指示灯亮。

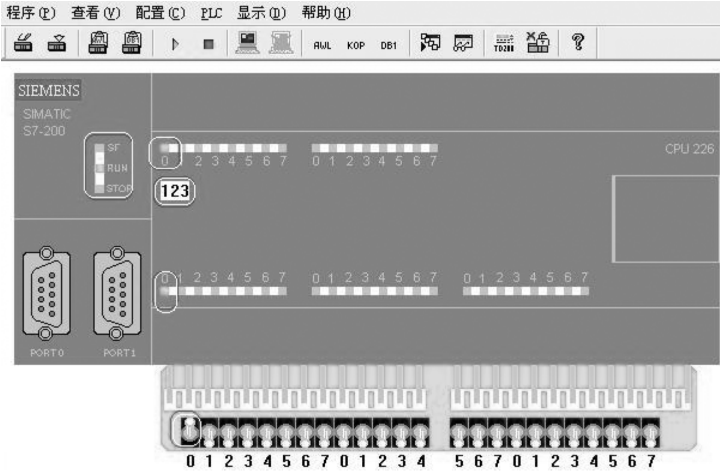


图 3-72　进行仿真

与 PLC 相比，仿真软件有省钱、方便等优势，但仿真软件毕竟不是真正的 PLC，它只具备 PLC 的部分功能，不能实现完全仿真。

小　　结

（1）重点掌握程序的编译、下载、调试和运行的全过程。
（2）通信不成功时解决方案的选择是本章难点。

习　　题

1. 计算机安装 STEP 7 - Micro/WIN 软件需要哪些软、硬件条件?
2. 没有 RS - 232C 接口的笔记本电脑要使用具备 RS - 232C 接口的编程电缆下载程序到

S7－200 PLC 中，应该做哪些预处理？

3. 在 STEP 7－Micro/WIN 软件中，“局部编译” 和 “完全编译” 的区别是什么？

4. 连接计算机的 RS－232C 接口和 PLC 的编程口之间的编程电缆时，为什么要关闭 PLC 的电源？

5. 当 S7－200 PLC 处于监控状态时，能否用软件设置 PLC 为 “停止” 模式？

6. 如何设置 CPU 的密码？怎样清除密码？怎样对整个工程加密？

7. 断电数据保持有几种形式实现？怎样判断数据块已经写入 EEPROM？

8. 状态表和趋势图有什么作用？怎样使用？二者有何联系？

9. 工具浏览条中有哪些重要的功能？

10. 交叉引用有什么作用？

11. 仿真软件的优缺点有哪些？

12. 某计算机的操作系统是 Windows 7，问此计算机上是否可以安装 STEP 7－Micro/WIN V4.0 SP6 软件？

13. 程序不能下载的可能原因是什么？

14. 用仿真软件 S7－200 SIM 2.0 仿真如图 3-73 所示梯形图。

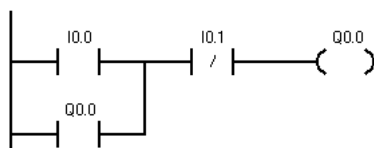


图 3-73　梯形图

15. 指出如图 3-74 所示梯形图的错误。

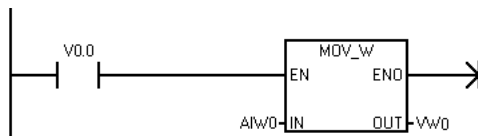


图 3-74　梯形图

第 4 章　PLC 的编程语言

本章主要介绍 S7－200 的数据类型；地址分配；S7－200 的编程语言。本章内容多，而且非常重要。

4.1　S7－200 PLC 的编程基础知识

4.1.1　数据的存储类型

1. 数制

（1）二进制

二进制数的 1 位（bit）只能取 0 和 1 两个不同的值，可以用来表示开关量的两种不同的状态，例如触点的断开和接通、线圈的通电和断电、灯的亮和灭等。在梯形图中，如果该位是 1 可以表示常开触点的闭合和线圈的得电，反之，该位是 0 可以表示常开触点的断开和线圈的断电。二进制用 2#表示，例如 2#1001 1101 1001 1101 就是 16 位二进制常数。十进制的运算规则是逢 10 进 1，二进制的运算规则是逢 2 进 1。

（2）十六进制

十六进制的 16 个数字是 0～9 和 A～F（对应于十进制中的 10～15），每个十六进制数字可用 4 位二进制表示，例如 16#A 用二进制表示为 2#1010。B#16#、W#16#、DW#16#分别表示十六进制的字节、字和双字。十六进制的运算规则是逢 16 进 1。学会二进制和十六进制之间的转化对于学习西门子 PLC 来说是十分重要的。

（3）BCD 码

BCD 码用 4 位二进制数（或者 1 位十六进制数）表示 1 位十进制数，例如 1 位十进制数 9 的 BCD 码是 1001。4 位二进制有 16 种组合，但 BCD 码只用到前十个，而后六个（1010～1111）没有在 BCD 码中使用。十进制的数字转换成 BCD 码是很容易的，例如十进制数 366 转换成十六进制 BCD 码则是 W#16#0366。

【关键点】十进制数 366 转换成十六进制数是 W#16#16E，这是要特别注意的。

BCD 码的最高 4 位二进制数用来表示符号，16 位 BCD 码字的范围是－999～＋999。32 位 BCD 码双字的范围是－9 999 999～＋9 999 999。不同数制的数的表示方法见表 4-1。

表 4-1　不同数制的数的表示方法

十进制	十六进制	二进制	BCD 码	十进制	十六进制	二进制	BCD 码
0	0	0000	00000000	3	3	0011	00000011
1	1	0001	00000001	4	4	0100	00000100
2	2	0010	00000010	5	5	0101	00000101

（续）

十进制	十六进制	二进制	BCD 码	十进制	十六进制	二进制	BCD 码
6	6	0110	00000110	11	B	1011	00010001
7	7	0111	00000111	12	C	1100	00010010
8	8	1000	00001000	13	D	1101	00010011
9	9	1001	00001001	14	E	1110	00010100
10	A	1010	00010000	15	F	1111	00010101

2. 数据的长度和类型

S7 - 200 将信息存于不同的存储器单元。每个单元都有唯一的地址，可以明确指出要存取的存储器地址。这就允许用户程序直接存取这个信息。表 4-2 列出了不同长度的数据所能表示的数值范围。

表 4-2　不同长度的数据表示的十进制和十六进制数范围

数据类型	数据长度	取值范围
字节（B）	8 位（1 字节）	0 ~ 255
字	16 位（2 字节）	0 ~ 65 535
位（bit）	1 位	0、1
整数（int）	16 位（2 字节）	0 ~ 65 535（无符号），-32 768 ~ +32 767（有符号）
双整数（dint）	32 位（4 字节）	0 ~ 4 294 967 295（无符号） -2 147 483 648 ~ +2 147 483 647（有符号）
双字（dword）	32 位（4 字节）	0 ~ 4 294 967 295
实数（real）	32 位（4 字节）	1. 175 495E - 38 ~ 3. 402 823E + 38（正数） -1. 175 495E - 38 ~ -3. 402 823E + 38（负数）
字符串（string）	8 位（1 字节）	

3. 常数

在 S7 - 200 的许多指令中都用到常数，常数有多种表示方法，如二进制、十进制和十六进制等。在表述二进制和十六进制时，要在数据前分别加“2#”或“16#”，格式如下：

二进制常数：2#1100，十六进制常数：16#234B1。其他的数据表述方法举例如下：

ASCII 码：“HELLOW”，实数：-3. 141 592 6，十进制数：234。

几个错误表示方法：八进制的“33”表示成“8#33”，十进制的“33”表示成“10# 33”，“2”用二进制表示成“2#2”，这些错误读者要避免。

若要存取存储区的某一位，则必须指定地址，包括存储器标识符、字节地址和位号。图 4-1 是一个位寻址的例子。其中，存储器标识符、字节地址（I 代表输入，2 代表字节 2）和位号之间用点号（. ）隔开。

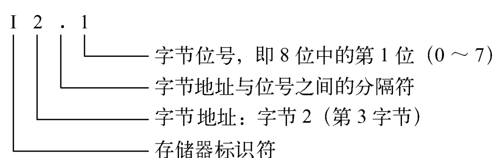


图 4-1　位寻址的例子

【例 4-1】 如图 4-2 所示，如果 MD0 = 16#

1F。那么，MB0、MB1、MB2 和 MB3 的数值是多少？M0.0 和 M3.0 是多少？

【解】

因为一个双字包含 4 个字节，一个字节包含 2 个十六进制位，所以 MD0 = 16#1F = 16#0000001F。根据图 4-2 可知，MB0 = 0，MB1 = 0，MB2 = 0，MB3 = 16#1F。由于 MB0 = 0，所以 M0.0 = 0。由于 MB3 = 16#1F = 2#00011111，所以 M3.0 = 1。这点不同于三菱 PLC，注意区分。



图 4-2　字节、字和双字的起始地址

【例 4-2】如图 4-3 所示的梯形图，请查看有无错误。

图 4-3　梯形图

【解】

这个程序从逻辑上看没有问题，但这个程序在实际运行时是有问题的。网络 1 是起停控制，当 V0.0 常开触点闭合后开始采集数据，而且 A-D 转换的结果存放在 VW0 中，VW0 包含 2 个字节 VB0 和 VB1，而 VB0 包含 8 个位，即 V0.0～V0.7。只要采集的数据经过 A-D 转换，造成 V0.0 位为 0，整个数据采集过程自动停止。初学者很容易犯类似的错误。读者可将 V0.0 改为 V2.0 即可，只要避开 VW0 中包含的 16 个位（V0.0～V0.7 和 V1.0～V1.7）都可行。

4.1.2　元件的功能与地址分配

1. 输入过程映像寄存器 I

输入过程映像寄存器与输入端相连，它是专门用来接收 PLC 外部开关信号的元件。在每次扫描周期的开始，CPU 对物理输入点进行采样，并将采样值写入输入过程映像寄存器中。可以按位、字节、字或双字来存取输入过程映像寄存器中的数据。输入过程映像寄存器 I0.0 的等效电路如图 4-4 所示，真实的回路中当按钮闭合，线圈 I0.0 得电，经过 PLC 内部

电路的转化，使得梯形图中常开触点 I0.0 闭合，理解这一点很重要。

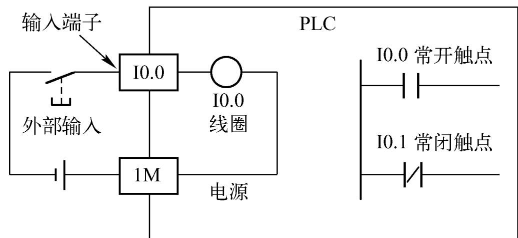


图 4-4　输入过程映像寄存器 I0.0 的等效电路

位格式：I[字节地址].[位地址]，如 I0.0。

字节、字或双字格式：I[长度][起始字节地址]，如 IB0、IW0、ID0。

2. 输出过程映像寄存器 Q

输出过程映像寄存器是用来将 PLC 内部信号输出传送给外部负载（用户输出设备）。输出过程映像寄存器线圈是由 PLC 内部程序的指令驱动，其线圈状态传送给输出单元，再由输出单元对应的硬触点来驱动外部负载。输出过程映像寄存器 Q0.0 的等效电路如图 4-5 所示。当梯形图中的线圈 Q0.0 得电，经过 PLC 内部电路的转化，使得真实回路中的常开触点 Q0.0 闭合，从而使得外部设备线圈得电，理解这一点很重要。

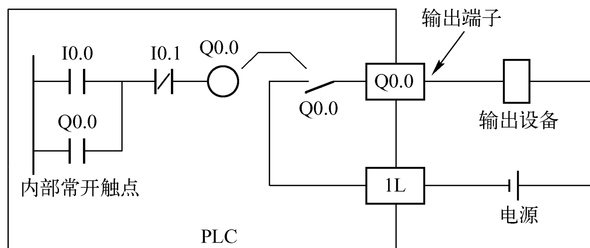


图 4-5　输出过程映像寄存器 Q0.0 的等效电路

在每次扫描周期的结尾，CPU 将输出过程映像寄存器中的数值复制到物理输出点上。可以按位、字节、字或双字来存取输出过程映像寄存器。

位格式：Q[字节地址].[位地址]，如 Q1.1。

字节、字或双字格式：Q[长度][起始字节地址]，如 QB5、QW5、QD5。

3. 变量存储器 V

可以用 V 存储器存储程序执行过程中控制逻辑操作的中间结果，也可以用它来保存与工序或任务相关的其他数据。变量存储器不能直接驱动外部负载。它可以按位、字节、字或双字来存取 V 存储区中的数据。

位格式：V[字节地址].[位地址]，如 V10.2。

字节、字或双字格式：V[长度][起始字节地址]，如 VB100、VW100、VD100。

4. 位存储器 M

位存储器是 PLC 中数量较多的一种继电器，一般的辅助继电器与继电器控制系统中的中间继电器相似。位存储器不能直接驱动外部负载，负载只能由输出过程映像寄存器的外部触点驱动。位存储器的常开与常闭触点在 PLC 内部编程时可无限次使用。可以用位存储区作为控

制继电器来存储中间操作状态和控制信息，并且可以按位、字节、字或双字来存取位存储区。

位格式：M[字节地址].[位地址]，如 M2.7。

字节、字或双字格式：M[长度][起始字节地址]，如 MB10、MW10、MD10。

注意：有的用户习惯使用 M 区作为中间地址，但 S7－200 CPU 中 M 区地址空间很小，只有 32 个字节，往往不够用。而 S7－200 CPU 中提供了大量的 V 区存储空间，即用户数据空间。V 存储区相对很大，其用法与 M 区相似，可以按位、字节、字或双字来存取 V 区数据，例如 V10.1、VB20、VW100、VD200 等。

【例 4-3】 图 4-6 所示的梯形图中，Q0.0 控制一盏灯，请分析当系统上电后接通 I0.0 和系统断电后又上电时灯的明暗情况。

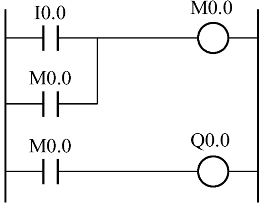


图 4-6 梯形图

【解】

当系统上电后接通 I0.0，Q0.0 线圈带电，并自锁，灯亮；系统断电后又上电，Q0.0 线圈处于断电状态，灯不亮。

5. 特殊存储器 SM

SM 位为 CPU 与用户程序之间传递信息提供了一种手段。可以用这些位选择和控制 S7－200 CPU 的一些特殊功能。例如，首次扫描标志位（SM0.1）、按照固定频率开关的标志位或者显示数学运算或操作指令状态的标志位，并且可以按位、字节、字或双字来存取 SM 位。

位格式：SM[字节地址].[位地址]，如 SM0.1。

节、字或者双字格式：SM[长度][起始字节地址]，如 SMB86、SMW22、SMD42。

特殊存储器的范围为 SM0～SM549，全部掌握是比较困难的，使用特殊存储器请参考有关手册，常用的特殊存储器字节 SMB0 见表 4-3。SM0.0、SM0.1、SM0.5 的时序图如图 4-7 所示。

表 4-3 特殊存储器字节 SMB0（SM0.0～SM0.7）

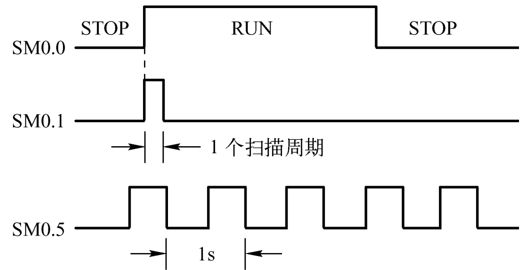
SM 位	描　述
SM0.0	该位始终为 1
SM0.1	该位在首次扫描时为 1，用途之一是调用初始化子程序
SM0.2	若保持数据丢失，则该位在一个扫描周期中为 1。该位可用做错误存储器位，或用来调用特殊启动顺序功能
SM0.3	开机后进入运行（RUN）方式，该位将被置 1 个扫描周期，该位可用做在启动操作之前给设备提供一个预热时间
SM0.4	该位提供一个时钟脉冲，30 s 为 1，30 s 为 0，周期为 1 min，它提供了一个简单易用的延时或 1 min 的时钟脉冲
SM0.5	该位提供一个时钟脉冲，0.5 s 为 1，0.5 s 为 0，周期为 1 s。它提供了一个简单易用的延时或 1 s 的时钟脉冲
SM0.6	该位为扫描时钟，本次扫描时置 1，下次扫描时置 0。可用做扫描计数器的输入
SM0.7	该位指示 CPU 工作方式开关的位置（0 为 TERM 位置，1 为 RUN 位置）。当开关在 RUN 位置时，用该位可使自由端口通信方式有效，那么当切换至 TERM 位置时，同编程设备的正常通信也会有效

【例 4-4】 图 4-8 所示的梯形图中，Q0.0 控制一盏灯，请分析当系统上电后灯的明暗情况。

【解】

因为 SM0.5 是周期为 1 s 的脉冲信号，所以灯亮 0.5 s，然后暗 0.5 s，以 1 s 为周期闪烁。

SM0.5 常用于报警灯的闪烁。



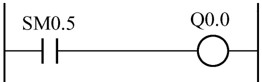


图 4-7　SM0.0、SM0.1、SM0.5 的时序图　　　　图 4-8　梯形图

6. 局部存储器 L

S7-200 有 64B 的局部存储器，其中 60B 可以用做临时存储器或者给子程序传递参数。如果用梯形图或功能块图编程，STEP 7-Micro/WIN 保留这些局部存储器的最后 4B。局部存储器和变量存储器 V 很相似，但只有一个区别：变量存储器是全局有效的，而局部存储器只在局部有效。全局是指同一个存储器可以被任何程序存取（包括主程序、子程序和中断服务程序），局部是指存储器区和特定的程序相关联。S7-200 给主程序分配 64B 的局部存储器，给每一级子程序嵌套分配 64B 的局部存储器，同样给中断服务程序分配 64B 的局部存储器。

子程序不能访问分配给主程序、中断服务程序或者其他子程序的局部存储器。同样，中断服务程序也不能访问分配给主程序或子程序的局部存储器。S7-200 PLC 根据需要分配局部存储器。也就是说，当主程序执行时，分配给子程序或中断服务程序的局部存储器是不存在的。当发生中断或者调用一个子程序时，需要分配局部存储器。新的局部存储器地址可能会覆盖另一个子程序或中断服务程序的局部存储器地址。

局部存储器在分配时 PLC 不进行初始化，初值可能是任意的。当在子程序调用中传递参数时，在被调用子程序的局部存储器中，由 CPU 替换其被传递的参数的值。局部存储器在参数传递过程中不传递值，在分配时不被初始化，可能包含任意数值。L 可以作为地址指针。

位格式：L[字节地址].[位地址]，如 L0.0。

字节、字或双字格式：L[长度][起始字节地址]，如 LB33。

下面的程序中，LD10 作为地址指针。

```
LD     SM0.0
MOVD   &VB0, LD10          //将 V 区的起始地址装载到指针中
```

7. 模拟量输入映像寄存器 AI

S7-200 将模拟量值（如温度或电压）转换成 1 个字长（16 位）的数字量。可以用区域标识符（AI）、数据长度（W）及字节的起始地址来存取这些值。因为模拟输入量为 1 个字长，并且从偶数位字节（如 0、2、4）开始，所以必须用偶数字节地址（如 AIW0、AIW2、AIW4）来存取这些值，如 AIW1 是错误的数据。模拟量输入值为只读数据。

格式：AIW[起始字节地址]，如 AIW0。以下为通道 0 模拟量输入的程序。

```
LD    SM0.0
MOVW  AIW0, MW10        //将通道0模拟量输入量转换为数字量后存入MW10中
```

8. 模拟量输出映像寄存器 AQ

S7－200 把 1 个字长的数字值按比例转换为电流或电压。可以用区域标识符（AQ）、数据长度（W）及字节的起始地址来改变这些值。因为模拟量为一个字长，且从偶数字节（如 0，2，4）开始，所以必须用偶数字节地址（如 AQW0、AQW2、AQW4）来改变这些值。模拟量输出值时只写数据。

格式：AQW[起始字节地址]，如 AQW0。以下为通道 0 模拟量输出的程序。

```
LD        SM0.0
MOVW      1234, AQW0        //将数字量1234转换成模拟量(如电压)从通道0输出
```

9. 定时器 T

在 S7－200 CPU 中，定时器可用于时间累计，其分辨率（时基增量）分为 1 ms、10 ms 和 100 ms 三种。定时器有以下两个变量。

- 当前值：16 位有符号整数，存储定时器所累计的时间。
- 定时器位：按照当前值和预置值的比较结果置位或者复位（预置值是定时器指令的一部分）。

可以用定时器地址来存取这两种形式的定时器数据。究竟使用哪种形式取决于所使用的指令：如果使用位操作指令，则是存取定时器位；如果使用字操作指令，则是存取定时器当前值。存取格式：T［定时器号］，如 T37。

S7－200 系列中定时器可分为接通延时定时器、有记忆的接通延时定时器和断开延时定时器三种。它们是通过对一定周期的时钟脉冲进行累计而实现定时的，时钟脉冲的周期（分辨率）有 1 ms、10 ms、100 ms 三种，当计数达到设定值时触点动作。

10. 计数器存储区 C

在 S7－200 CPU 中，计数器可以用于累计其输入端脉冲电平由低到高的次数。CPU 提供了三种类型的计数器，一种只能增加计数；一种只能减少计数；另外一种既可以增加计数，又可以减少计数。计数器有以下两种形式。

- 当前值：16 位有符号整数，存储累计值。
- 计数器位：按照当前值和预置值的比较结果置位或者复位（预置值是计数器指令的一部分）。

可以用计数器地址来存取这两种形式的计数器数据。究竟使用哪种形式取决于所使用的指令：如果使用位操作指令，则是存取计数器位；如果使用字操作指令，则是存取计数器当前值。存取格式：C［计数器号］，如 C24。

11. 高速计数器 HC

高速计数器用于对高速事件计数，它独立于 CPU 的扫描周期。高速计数器有一个 32 位的有符号整数计数值（或当前值）。若要存取高速计数器中的值，则应给出高速计数器的地址，即存储器类型（HC）加上计数器号（如 HC0）。高速计数器的当前值是只读数据，仅可以作为双字（32 位）来寻址。

格式：HC[高速计数器号]，如 HC1。

12. 累加器 AC

累加器是可以像存储器一样使用的读写设备。例如，可以用它来向子程序传递参数，也可以从子程序返回参数，以及用来存储计算的中间结果。S7－200 提供 4 个 32 位累加器（AC0、AC1、AC2 和 AC3），并且可以按字节、字或双字的形式来存取累加器中的数值。

被访问的数据长度取决于存取累加器时所使用的指令。当以字节或者字的形式存取累加器时，使用的是数值的低 8 位或低 16 位。当以双字的形式存取累加器时，使用全部 32 位。

格式：AC[累加器号]，如 AC0。以下为将常数 18 移入 AC0 中的程序。

```
LD    SM0.0
MOV   B18,AC0          //将常数 18 移入 AC0
```

13. 顺控继电器存储 S

顺控继电器位（S）用于组织机器操作或者进入等效网路的步骤。SCR 提供控制程序的逻辑分段。可以按位、字节、字或双字来存取 S 位。

位：S[字节地址].[位地址]，如 S3.1。

字节、字或者双字：S[长度][起始字节地址]。

4.1.3 STEP 7 中的编程语言

STEP 7 中有梯形图、语句表和功能块图 3 种基本编程语言，可以相互转换。通过安装软件包，还有其他的编程语言，以下简要介绍。

（1）顺序功能图（SFC）

STEP 7 中为 S7－Graph，不是 STEP 7 的标准配置，需要安装软件包，是针对顺序控制系统进行编程的图形编程语言，特别适合顺序控制程序编写。

（2）梯形图（LAD）

直观易懂，适合于数字量逻辑控制。“能流”（Power flow）与程序执行的方向一致。梯形图适合于熟悉继电器电路的人员使用。设计复杂的触点电路时最好用梯形图。其应用最为广泛。

（3）语句表（STL）

功能比梯形图或功能块图强。语句表可供喜欢用汇编语言编程的用户使用。语句表的输入快，可以在每条语句后面加上注释。设计高级应用程序时建议使用语句表。

（4）功能块图（FBD）

“LOGO!”系列微型 PLC 使用功能块图编程。功能块图适合于熟悉数字电路的人使用。

（5）结构文本（ST）

STEP 7 的 S7－SCL（结构化控制语言）符合 EN 61131－3 标准。SCL 适合于复杂的公式计算、复杂的计算任务和最优化算法，或管理大量的数据等。S7－SCL 编程语言适合于熟悉高级编程语言（例如 PASCAL 或 C 语言）的人使用。S7－200 不支持此功能。

（6）S7－HiGraph 编程语言

图形编程语言 S7－HiGraph 属于可选软件包，它用状态图（stategraphs）来描述异步、非顺序过程的编程语言。HiGraph 适合于异步非顺序过程的编程。S7－200 不支持此功能。

（7）S7 CFC 编程语言

可选软件包 CFC（Continuous Function Chart，连续功能图）用图形方式连接程序库中以块的形式提供的各种功能。CFC 适合于连续过程控制的编程。它不是 STEP 7 的标准配置，需要安装软件包。S7 -200 不支持此功能。

在 STEP 7 编程软件中，如果程序块没有错误，并且被正确地划分为网络，在梯形图、功能块图和语句表之间可以转换。如果部分网络不能转换，则用语句表表示。

4.2 位逻辑指令

基本逻辑指令是指构成基本逻辑运算功能指令的集合，包括基本位操作、置位/复位、边沿触发、逻辑栈、定时、计数、比较等逻辑指令。S7 -200 系列 PLC 共有 27 条逻辑指令，现按用途分类如下。

4.2.1 基本位操作指令

1. 装载及线圈驱动指令

LD（Load）：常开触点逻辑运算开始。

LDN（Load Not）：常闭触点逻辑运算开始。

=（Out）：线圈驱动。

图 4-9 所示梯形图及语句表表示上述三条指令的用法。

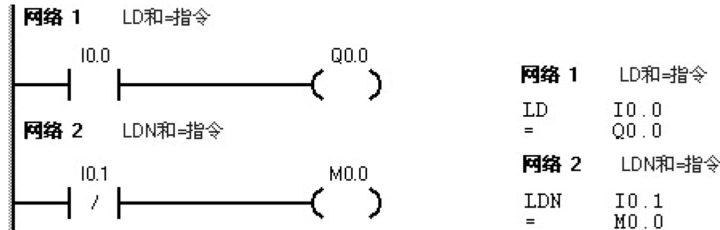


图 4-9 LD、LDN、= 指令应用举例

装载及线圈驱动指令使用说明：

① LD（Load）：装载指令，对应梯形图从左侧母线开始，连接常开触点。

② LDN（Load Not）：装载指令，对应梯形图从左侧母线开始，连接常闭触点。

③ =（Out）：线圈输出指令，可用于输出过程映像寄存器、辅助继电器、定时器及计数器等，但不能用于输入过程映像寄存器。

④ LD、LDN 的操作数：I、Q、M、SM、T、C、S。= 的操作数：Q、M、SM、T、C、S。

图 4-9 中梯形图的含义解释：当网络 1 中的常开触点 I0.0 接通，则线圈 Q0.0 得电；当网络 2 中的常闭触点 I0.1 接通，则线圈 M0.0 得电。此梯形图的含义与以前学过的电气控制中的电气图类似。

2. 触点串联指令（图 4-10 所示梯形图及指令表表示了上述两条指令的用法）

A（And）：常开触点串联。

AN（And Not）：常闭触点串联。

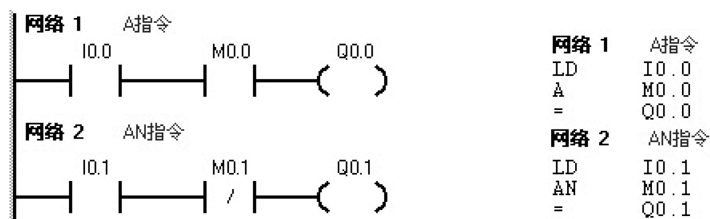


图 4-10　A、AN 指令应用举例

触点串联指令使用说明：

① A、AN：与操作指令，是单个触点串联指令，可连续使用。

② A、AN 的操作数：I、Q、M、SM、T、C、S。

图 4-10 中梯形图的含义解释：当网络 1 中的常开触点 I0.0、M0.0 同时接通，则线圈 Q0.0 得电，常开触点 I0.0、M0.0 都不接通，或者只有一个接通，线圈 Q0.0 不得电，常开触点 I0.0、M0.0 是串联（与）关系。当网络 2 中的常开触点 I0.1、常闭触点 M0.1 同时接通，则线圈 Q0.1 得电，常开触点 I0.1 和常闭触点 M0.1 是串联（与非）关系。

3. 触点并联指令

O（Or）：常开触点并联。

ON（Or Not）：常闭触点并联。

图 4-11 所示梯形图及指令表表示了上述两条指令的用法。

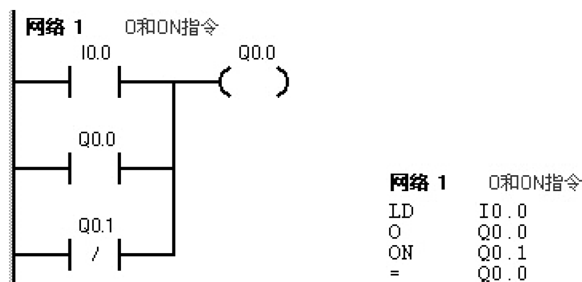


图 4-11　O、ON 指令应用举例

① O、ON：或操作指令，是单个触点并联指令，可连续使用。

② O、ON 的操作数：I、Q、M、SM、T、C、S。

图 4-11 中梯形图的含义解释：当网络 1 中的常开触点 I0.0、Q0.0，常闭触点 Q0.1 有一个或者多个接通，则线圈 Q0.0 得电，常开触点 I0.0、Q0.0 和常闭触点 Q0.1 是并联（或、或非）关系。

4. 并联电路块的串联指令

ALD（And Load）：并联电路块的串联连接。

图 4-12 表示了 ALD 指令的用法。

并联电路块的串联指令使用说明：

① 并联电路块与前面电路串联时，使用 ALD 指令。电路块的起点用 LD 或 LDN 指令，并联电路块结束后，使用 ALD 指令与前面电路块串联。

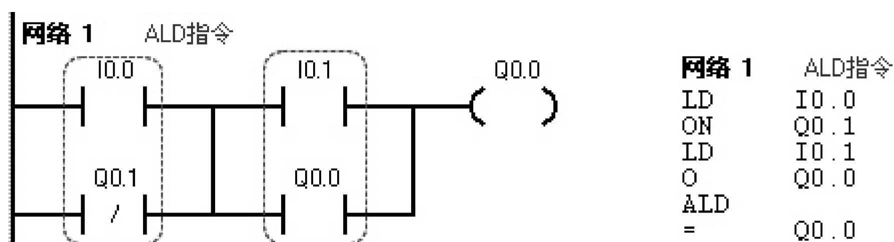


图 4-12　ALD 指令应用举例

② ALD 无操作数。

图 4-12 中梯形图的含义解释：实际上就是把第一个虚线框中的触点 I0.0 和触点 Q0.1 并联，再将第二个虚线框中的触点 I0.1 和触点 Q0.0 并联，最后把两个虚线框中并联后的结果串联。

5. 电路块的并联指令

OLD（Or Load）：串联电路块的并联连接。图 4-13 表示了 OLD 指令的用法。

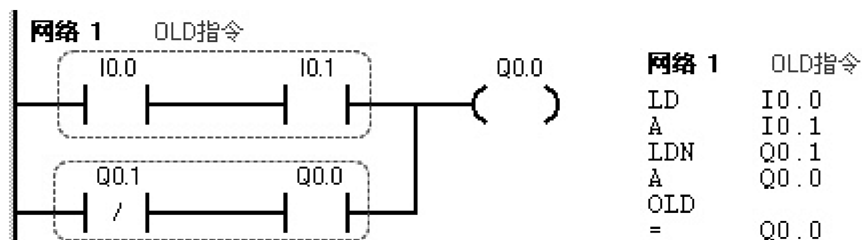


图 4-13　OLD 指令应用举例

串联电路块的并联指令使用说明：

① 串联电路块并联连接时，其支路的起点均以 LD 或 LDN 开始，终点以 OLD 结束。

② OLD 无操作数。

图 4-13 中梯形图的含义解释：实际上就是把第一个虚线框中的触点 I0.0 和触点 I0.1 串联，再将第二个虚线框中的触点 Q0.1 和触点 Q0.0 串联，最后把两个虚线框中串联后的结果并联。

如图 4-14 所示是 OLD 和 ALD 指令的使用。

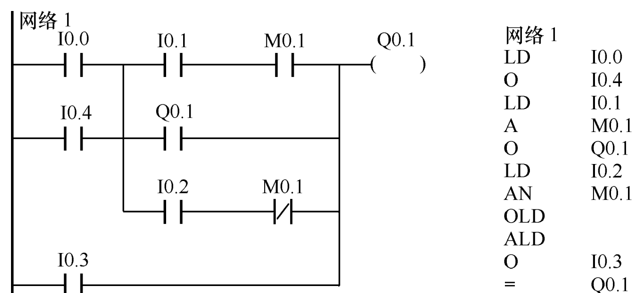


图 4-14　OLD、ALD 指令的使用

4.2.2 置位/复位指令

普通线圈获得能量流时，线圈通电（存储器位置 1），能量流不能到达时，线圈断电（存储器位置 0）。置位/复位指令将线圈设计成置位线圈和复位线圈两大部分。置位线圈受到脉冲前沿触发时，线圈通电锁存（存储器位置 1），复位线圈受到脉冲前沿触发时，线圈断电锁存（存储器位置 0），下次置位、复位操作信号到来前，线圈状态保持不变（自锁）。置位/复位指令格式见表 4-4。

表 4-4 置位/复位指令格式

LAD	STL	功 能
S-BIT ——(S) N	S S-BIT，N	从起始位（S-BIT）开始的 *N* 个元件置 1 并保持
S-BIT ——(R) N	R S-BIT，N	从起始位（S-BIT）开始的 *N* 个元件清 0 并保持

R、S 指令的使用如图 4-15 所示，当 PLC 上电时，Q0.0 和 Q0.1 都通电，当 I0.1 接通时，Q0.0 和 Q0.1 都断电。

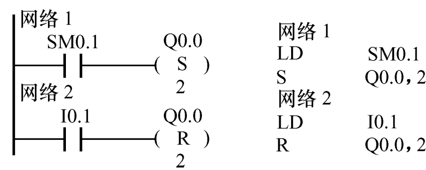


图 4-15 R、S 指令的使用

【关键点】 编程时，置位、复位线圈之间间隔的网络个数可以任意设置。置位、复位线圈通常成对使用，也可单独使用。

4.2.3 RS 触发器指令

RS 触发器具有置位与复位的双重功能，RS 触发器是复位优先，当置位（S）和复位（R）同时为真时，输出为假。而 SR 触发器是置位优先触发器，当置位（S）和复位（R）同时为真时，输出为真。RS 触发指令应用如图 4-16 所示。

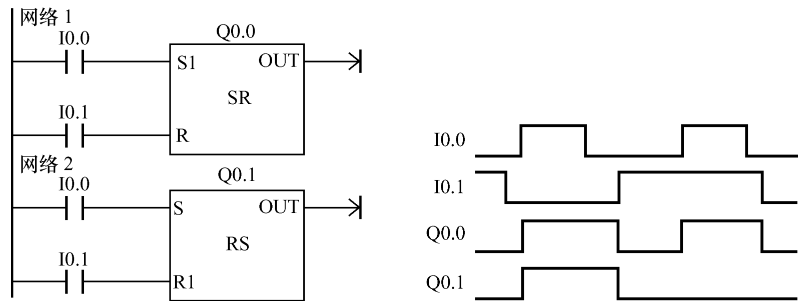


图 4-16 RS 触发指令应用

【例 4-5】抢答器有 I0.0、I0.1 和 I0.2 三个输入，对应的输出为 Q0.0、Q0.1 和 Q0.2，输入复位为 I0.3。要求三个人任意抢答，先按动瞬时按钮的指示灯优先亮，且只能亮一盏灯，主持人按复位按钮后，抢答重新开始。请编写相关程序实现此功能。

【解】程序如图 4-17 所示。

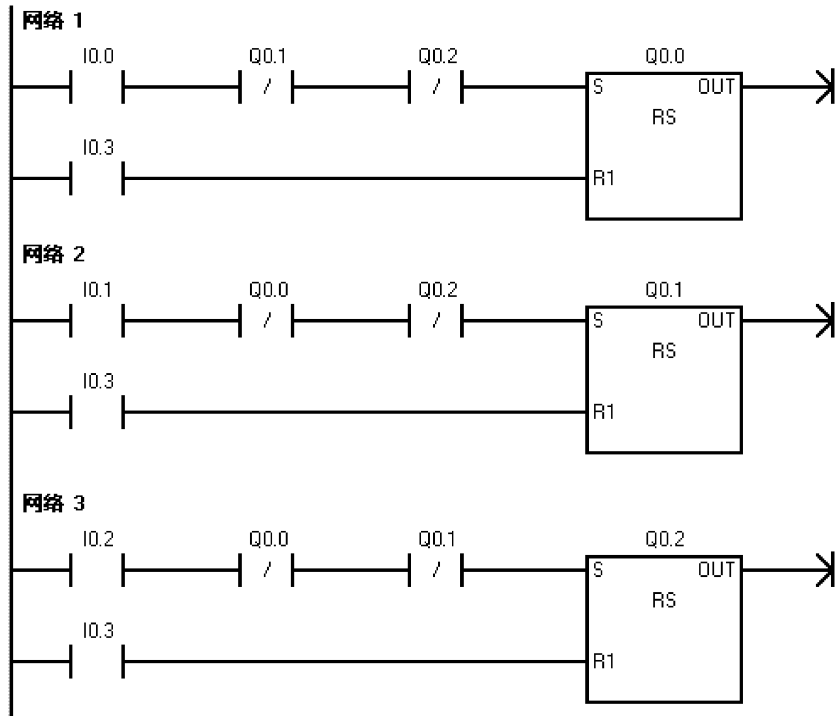


图 4-17　RS 触发指令应用举例

【例 4-6】设计一个程序，实现用一个单按钮控制一盏灯的亮和灭，即按奇数次按钮灯亮，按偶数次按钮灯灭。

【解】

最简单的方法是用 SR 或者 RS 指令编写程序。方法一：先用 SR 指令，程序如图 4-18 所示。当第一次按下 I0.0 按钮时，Q0.0 置位，当第二次按下 I0.0 按钮时，Q0.0 复位，灯灭。

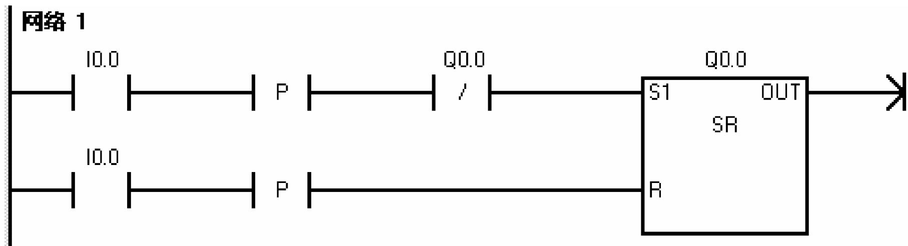


图 4-18　SR 触发指令应用举例

方法二：这个题目还有另一种解法，就是用 RS 指令，梯形图如图 4-19 所示，当第一次按下按钮时，Q0.0 线圈得电（灯亮），Q0.0 常开触点闭合，当第二次按下按钮时，S 和

R1 端子同时获得高电平，由于复位优先，所以 Q0.0 线圈断电（灯灭）。

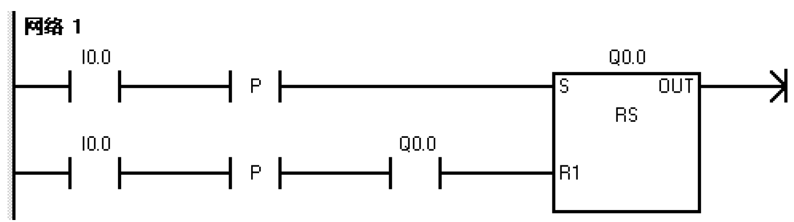


图 4-19　RS 触发指令应用举例

4.2.4　边沿触发指令

边沿触发是指用边沿触发信号产生一个机器周期的扫描脉冲，通常用做脉冲整形。边沿触发指令分为正跳变（上升沿）触发和负跳变（下降沿）触发两大类。正跳变触发指输入脉冲的上升沿使触点闭合（ON）一个扫描周期。负跳变触发指输入脉冲的下降沿使触点闭合（ON）一个扫描周期。边沿触发指令格式见表 4-5。

表 4-5　边沿触发指令格式

LAD	STL	功　　能
─┤P├─	EU	正跳变，无操作元件
─┤N├─	ED	负跳变，无操作元件

【例 4-7】 如图 4-20 所示的程序，若 I0.0 上电一段时间后再断开，请画出 I0.0、Q0.0、Q0.1 和 Q0.2 的时序图。

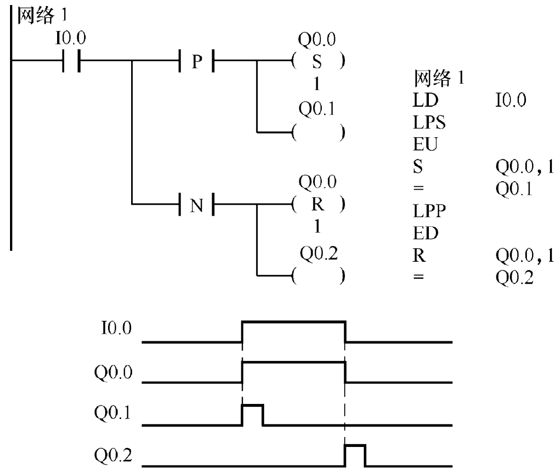


图 4-20　边沿触发指令应用示例

【解】

如图 4-20 所示，在 I0.0 的上升沿，触点（EU）产生一个扫描周期的时钟脉冲，驱动

输出线圈 Q0.1 通电一个扫描周期，Q0.0 通电，并使输出线圈 Q0.0 置位并保持。

在 I0.0 的下降沿，触点（ED）产生一个扫描周期的时钟脉冲，驱动输出线圈 Q0.2 通电一个扫描周期，并使输出线圈 Q0.0 复位并保持。

【例 4-8】 设计一个程序，实现用一个单按钮控制一盏灯的亮和灭，即按奇数次按钮灯亮，按偶数次按钮灯灭。

【解】

当 I0.0 第一次闭合时，V0.0 接通一个扫描周期，使得 Q0.0 线圈得电一个扫描周期，当下一次扫描周期到达，Q0.0 常开触点闭合并自锁，灯亮。

当 I0.0 第二次闭合时，V0.0 接通一个扫描周期，使得 Q0.0 线圈得电一个扫描周期，切断 Q0.0 的常开触点和 V0.0 的常开触点，使得灯灭。梯形图如图 4-21 所示。

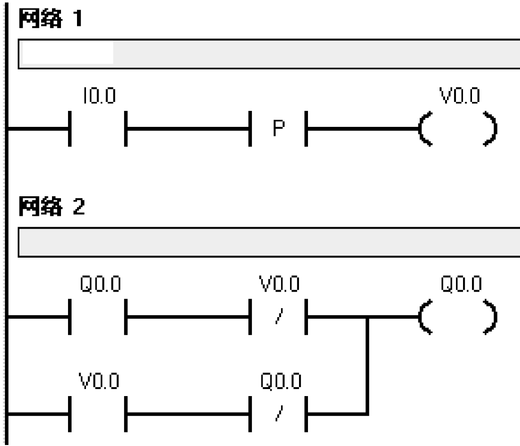


图 4-21　梯形图

【例 4-9】 用西门子 S7-200 PLC 对一台三相异步电动机进行“正-停-反”控制，请设计电气原理图，编写梯形图指令。

【解】

三相异步电动的“正-停-反”控制，类似于使用二次三相异步电动机的“起停”控制，电气原理图和梯形图如图 4-22 和图 4-23 所示。

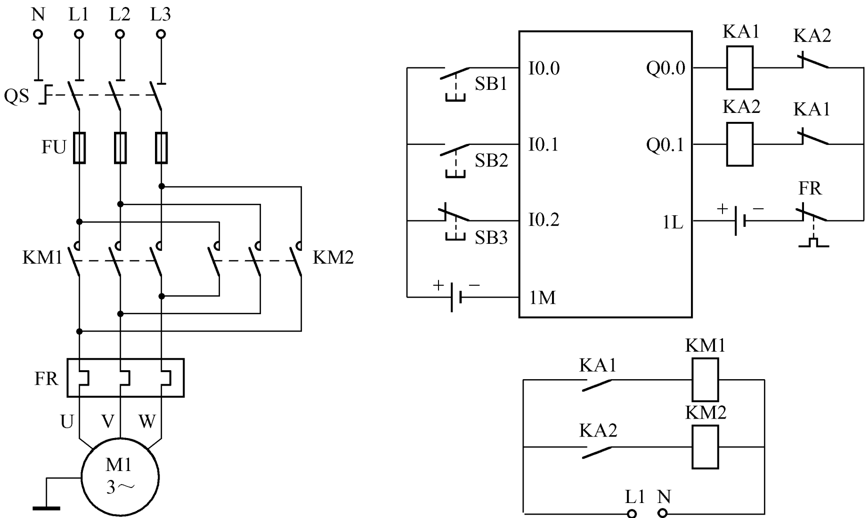


图 4-22　电气原理图

【关键点】 SB3 是停止按钮，应该接常闭触点，这主要基于安全考虑，这点读者务必注意。由于 SB3 接常闭触点，所以在梯形图中，I0.2 要用常开触点，这点初学者容易出错。

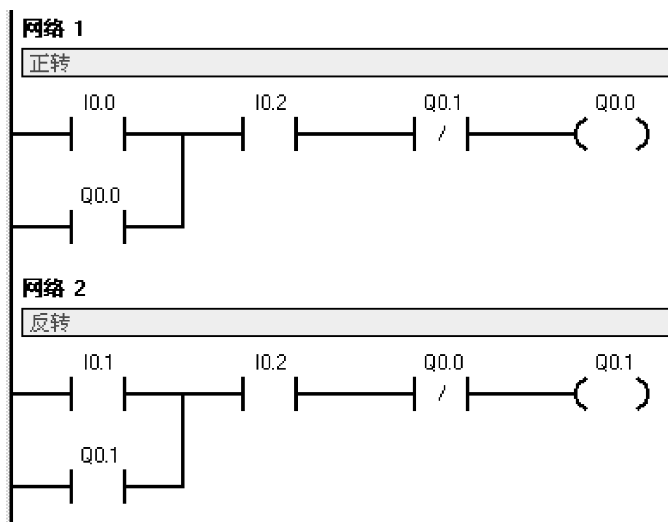


图 4-23 “正 - 停 - 反”梯形图

4. 2. 5 逻辑栈操作指令

LD 装载指令是从梯形图最左侧的母线画起的，如果要生成一条分支的母线，则需要利用语句表的栈操作指令来描述。

栈操作语句表指令格式如下：

LPS：逻辑堆栈指令，即把栈顶值复制后压入堆栈，栈底值丢失。

LRD：逻辑读栈指令，即把逻辑堆栈第二级的值复制到栈顶，堆栈没有压入和弹出。

LPP：逻辑弹栈指令，即把堆栈弹出一级，原来第二级的值变为新的栈顶值。

图 4-24 所示为逻辑栈操作指令对栈区的影响，图中 iv. x 表示存储在栈区某个程序断点的地址。

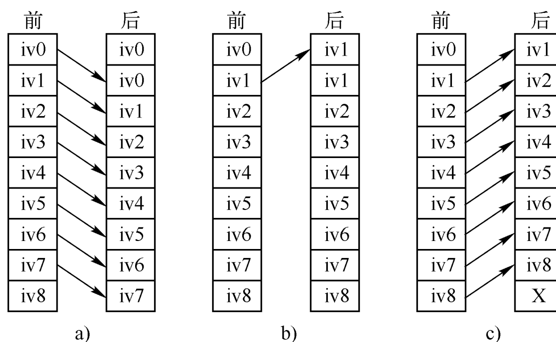


图 4-24 栈操作指令的操作过程

图 4-25 所示的例子说明了这几条指令的作用。其中只用了 2 层栈，实际上逻辑堆栈有 9 层，故可以连续使用多次 LPS 指令。但要注意 LPS 和 LPP 必须配对使用。

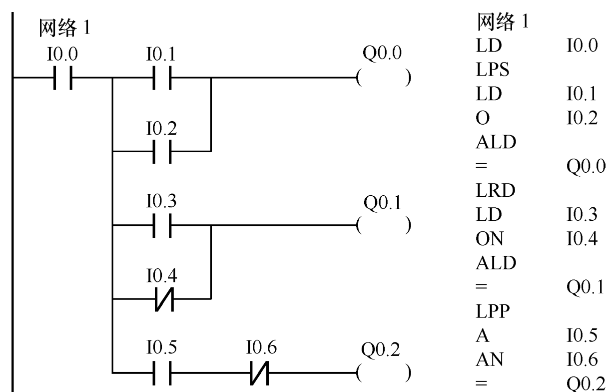


图 4-25　LPS、LRD、LPP 指令应用示例

4.3　定时器与计数器指令

4.3.1　定时器指令

S7-200 PLC 的定时器为增量型定时器，用于实现时间控制，可以按照工作方式和时间基准分类。

1. 工作方式

按照工作方式，定时器可分为通电延时型（TON）、有记忆的通电延时型或保持型（TONR）、断电延时型（TOF）三种类型。

2. 时间基准

按照时间基准（简称时基），定时器可分为 1 ms、10 ms、100 ms 三种类型，时间基准不同，定时精度、定时范围和定时器的刷新方式也不同。

定时器的工作原理为定时器的使能端输入有效后，当前值寄存器对 PLC 内部的时基脉冲增 1 计数，最小计时单位为时基脉冲的宽度。故时间基准代表着定时器的定时精度（分辨率）。

定时器的使能端输入有效后，当前值寄存器对时基脉冲递增计数，当计数值大于或等于定时器的预置值后，状态位置 1。从定时器输入有效到状态位置 1 经过的时间称为定时时间。定时时间等于时基乘以预置值，时基越大，定时的时间越长，但精度越差。

1 ms 定时器每隔 1 ms 刷新一次，与扫描周期和程序处理无关。因而当扫描周期较长时，定时器在一个周期内可能被多次刷新，其当前值在一个扫描周期内不一定保持一致。

10 ms 定时器在每个扫描周期开始时自动刷新。由于每个扫描周期只刷新一次，故在每次程序处理期间，其当前值为常数。

100 ms 定时器在定时器指令执行时被刷新，下一条执行的指令即可使用刷新后的结果，使用方便可靠。但应当注意，如果定时器的指令不是每个周期都执行（条件跳转时），定时器就不能及时刷新，可能会导致出错。

CPU 22X PLC 的 256 个定时器分属 TON（TOF）和 TONR 工作方式，以及 3 种时基标准（TON 和 TOF 共享同一组定时器，不能重复使用，例如当 T37 用于 TON 工作方式，就不可

以用于 TOF 工作方式）。其详细分类方法见表 4-6。

表 4-6　定时器工作方式及类型

工 作 方 式	时间基准/ms	最大定时时间/s	定时器型号
TONR	1	32.767	T0，T64
	10	327.67	T1～T4，T65～T68
	100	3276.7	T5～T31，T69～T95
TON/TOF	1	32.767	T32，T96
	10	327.67	T33～T36，T97～T100
	100	3276.7	T37～T63，T101～T255

3. 工作原理分析

下面分别叙述 TON、TONR、TOF 3 种类型定时器的使用方法。这 3 类定时器均有使能输入端 IN 和预置值输入端 PT。PT 预置值的数据类型为 INT，最大预置值是 32767。

（1）通电延时型定时器（TON）

使能端（IN）输入有效时，定时器开始计时，当前值从 0 开始递增，大于或等于预置值（PT）时，定时器输出状态位置 1。使能端输入无效（断开）时，定时器复位（当前值清 0，输出状态位置 0）。通电延时型定时器指令和参数见表 4-7。

表 4-7　通电延时型定时器指令和参数

LAD	参　　数	数 据 类 型	说　　明	存　储　区
Txxx IN　TON PT　PT　???ms	T xxx	WORD	表示要启动的定时器号	T32，T96，T33～T36，T97～T100，T37～T63，T101～T255
	PT	INT	定时器时间值	I，Q，M，D，L，T，S，SM，AI，T，C，AC，常数，＊VD，＊LD，＊AC
	IN	BOOL	使能	I，Q，M，SM，T，C，V，S，L

【例 4-10】 已知梯形图和 I0.1 时序如图 4-26 所示，请画出 Q0.0 的时序图。

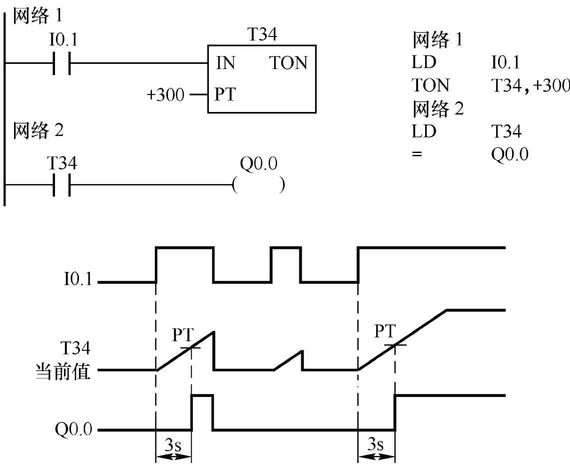


图 4-26　通电延时型定时器应用示例

【解】

当接通 I0.1，延时 3 s 后，Q0.0 得电。

【例 4-11】设计一个延时脉冲产生程序，延时时间是 5 s，脉冲的时序图如图 4-27 所示。

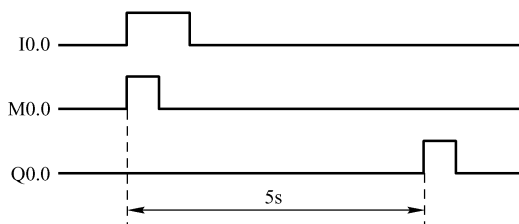


图 4-27　时序图

【解】

程序如图 4-28 所示。

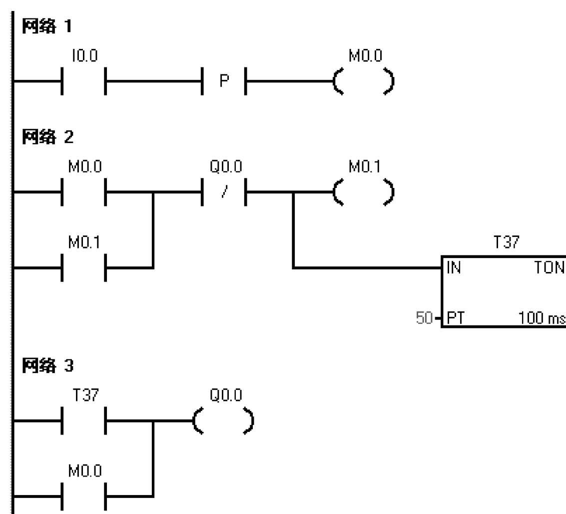


图 4-28　程序

【例 4-12】设计延时接通，延时断开程序，时序图如图 4-29 所示。

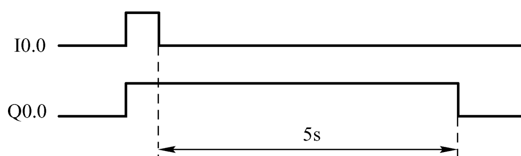


图 4-29　时序图

【解】

程序如图 4-30 所示。

【例 4-13】设计一段程序，实现一盏灯亮 3 s，灭 3 s，不断循环，且能实现起停控制。

【解】当接通 SB1 按钮，灯 HL1 亮，T37 延时 3 s 后，灯 HL1 灭，T38 延时 3 s 后，切断

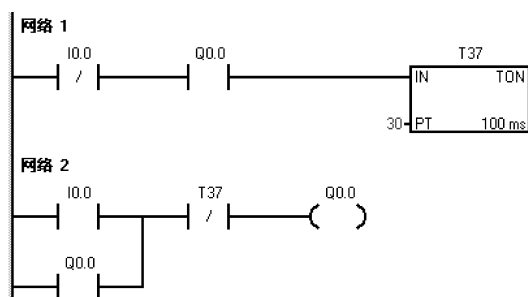


图 4-30　程序

T37，灯 HL1 亮，如此循环。接线图如图 4-31 所示，梯形图如图 4-32 所示。

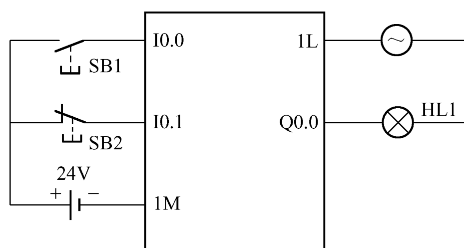


图 4-31　接线图

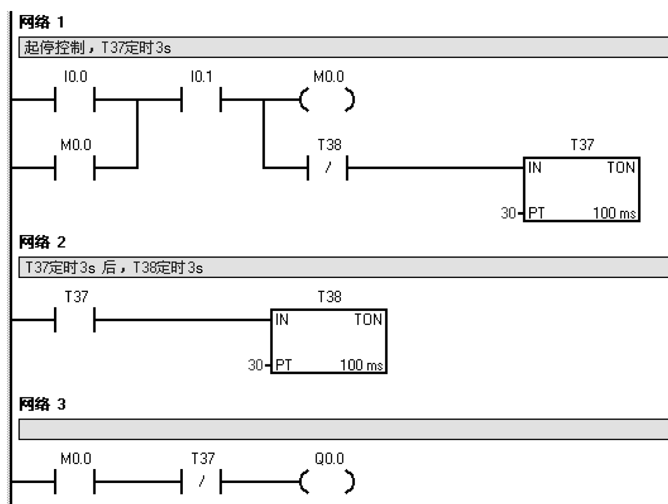


图 4-32　梯形图

【**例 4-14**】脉冲宽度可控电路时序图如图 4-33 所示，请设计程序。

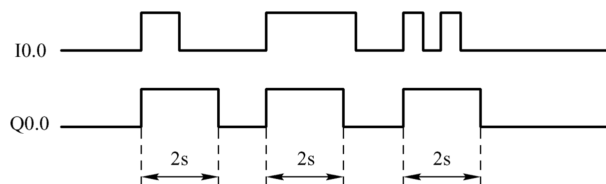


图 4-33　时序图

【解】

程序如图 4–34 所示。

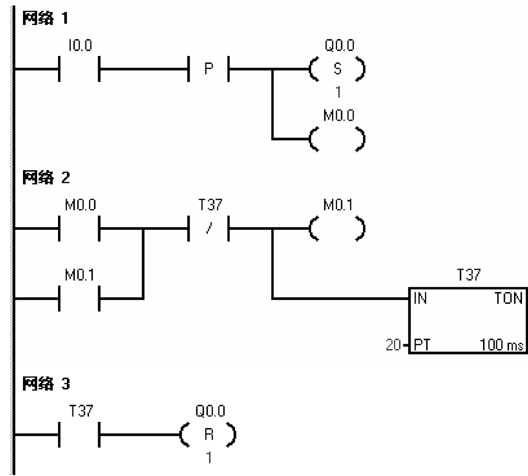


图 4–34　程序

（2）有记忆的通电延时型定时器（TONR）

使能端输入有效时，定时器开始计时，当前值递增，当前值大于或等于预置值时，输出状态位置 1。使能端输入无效时，当前值保持（记忆），使能端再次接通有效时，在原记忆值的基础上递增计时。有记忆的通电延时型定时器采用线圈的复位指令进行复位操作，当复位线圈有效时，定时器当前值清 0，输出状态位置 0。有记忆的通电延时型定时器指令和参数见表 4–8。

表 4–8　有记忆的通电延时型定时器指令和参数

LAD	参　　数	数 据 类 型	说　　明	存　储　区
Txxx IN　TONR PT - PT　??? ms	T xxx	WORD	表示要启动的定时器号	T0，T64，T1 ~ T4，T65 ~ T68，T5 ~ T31，T69 ~ T95
	PT	INT	定时器时间值	I，Q，M，D，L，T，S，SM，AI，T，C，AC，常数，＊VD，＊LD，＊AC
	IN	BOOL	使能	I，Q，M，SM，T，C，V，S，L

【例 4–15】 已知梯形图以及 I0. 0 和 I0. 1 的时序如图 4–35 所示，请画出 Q0. 0 的时序图。

【解】

当接通 I0. 0，延时 1s 后，Q0. 0 得电；I0. 0 断电后，Q0. 0 仍然保持得电。当 I0. 1 接通时，定时器复位，Q0. 0 断电。

【关键点】 有记忆的通电延时型定时器的线圈带电后，必须复位才能断电。

（3）断电延时型定时器（TOF）

使能端输入有效时，定时器输出状态位立即置 1，当前值清 0。使能端断开时，开始计时，当前值从 0 递增，当前值达到预置值时，定时器输出状态位复位置 0，并停止计时，当前值保持。断电延时型定时器指令和参数见表 4–9。

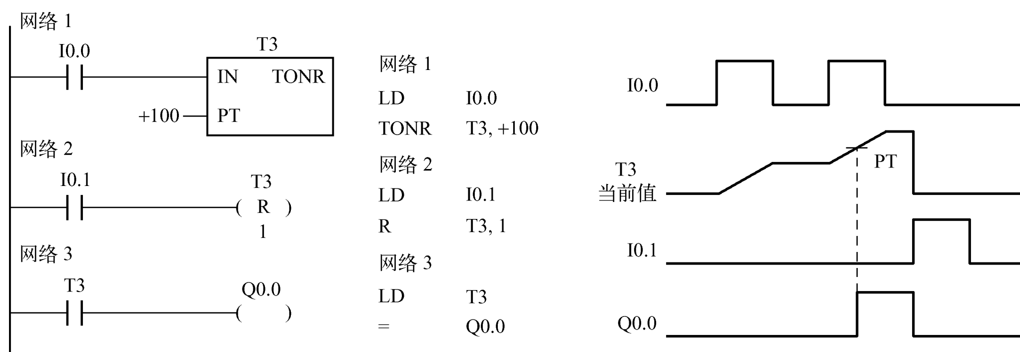


图 4-35　有记忆的通电型延时定时器应用示例

表 4-9　断电延时型定时器指令和参数

LAD	参　数	数据类型	说　明	存　储　区
Txxx IN TOF PT - PT ???ms	T xxx	WORD	表示要启动的定时器号	T32，T96，T33 ~ T36，T97 ~ T100，T37 ~ T63，T101 ~ T255
	PT	INT	定时器时间值	I，Q，M，D，L，T，S，SM，AI，T，C，AC，常数，* VD，* LD，* AC
	IN	BOOL	使能	I，Q，M，SM，T，C，V，S，L

【例 4-16】 已知梯形图以及 I0.0 的时序如图 4-36 所示，请画出 Q0.0 的时序图。

【解】 当接通 I0.0，Q0.0 得电；I0.0 断电 5s 后，Q0.0 失电。

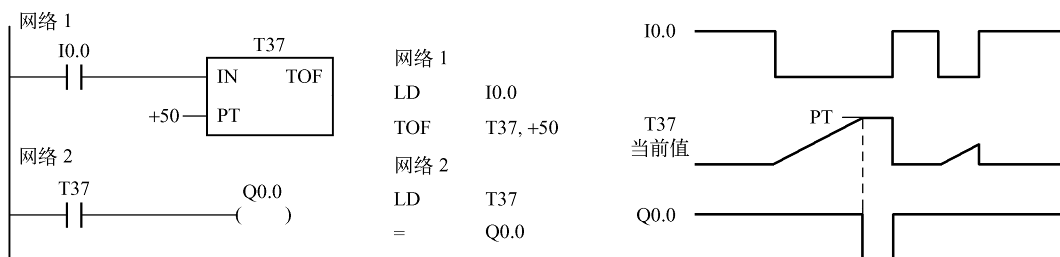


图 4-36　断电延时型定时器应用示例

【例 4-17】 某车库中有一盏灯，当人离开车库后，按下停止按钮，5 s 后灯熄灭，请编写程序。

【解】 当接通 SB1 按钮，灯 HL1 亮；按下 SB2 按钮 5 s 后，灯 HL1 灭。接线图如图 4-37 所示，梯形图如图 4-38 所示。

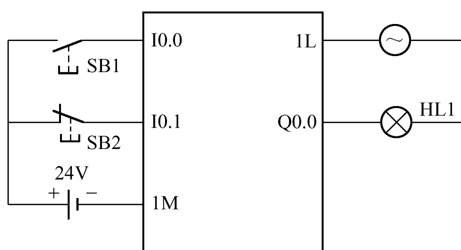


图 4-37　接线图

【例 4-18】 鼓风机系统一般由引风机和鼓风机两级构成。当按下起动按钮之后，引风机先工作，工作 5 s 后，鼓风机工作。按下停止按钮之后，鼓风机先停止工作，5 s 之后，引风机才停止工作。

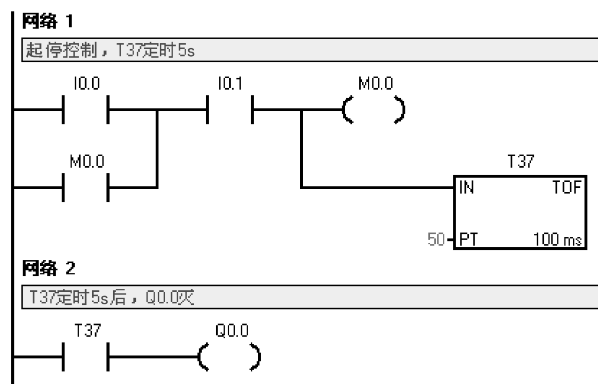


图 4-38　梯形图

【解】

1）PLC 的 I/O 分配。PLC 的 I/O 分配见表 4-10。

表 4-10　PLC 的 I/O 分配表

输　入			输　出		
名　称	符　号	输 入 点	名　称	符　号	输 出 点
开始按钮	SB1	I0. 0	鼓风机	KA1	Q0. 0
停止按钮	SB2	I0. 1	引风机	KA2	Q0. 1

2）控制系统的接线。鼓风机控制系统的接线比较简单，如图 4-39 所示。

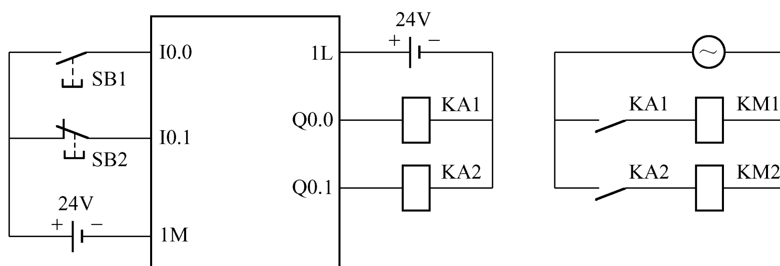


图 4-39　PLC 接线图

3）编写程序。引风机在按下停止按钮后还要运行 5 s，容易想到要使用 TOF 定时器；鼓风机在引风机工作 5 s 后才开始工作，因而容易想到使用 TON 定时器，不难设计梯形图，如图 4-40 所示。

【例 4-19】 常见的小区门禁，用来阻止陌生车辆直接出入。要求编写门禁系统控制程序实现如下控制功能。小区保安可以手动控制门开，到达门开限位开关时，停止 20 s 后自动关闭，在关闭过程中如果检测到有人通过（用一个按钮模拟），则停止 5 s，然后继续关闭，到达门关限位开关时停止。

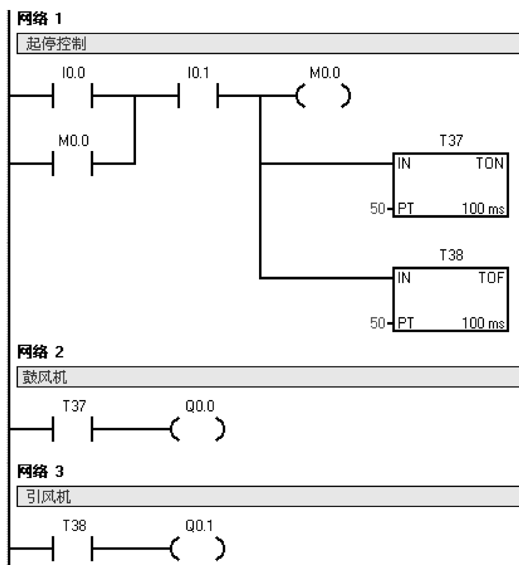


图 4-40　鼓风机控制梯形图

【解】

1）PLC 的 I/O 分配。PLC 的 I/O 分配见表 4-11。

表 4-11　PLC 的 I/O 分配表

输　入			输　出		
名　称	符　号	输　入　点	名　称	符　号	输　出　点
开始按钮	SB1	I0. 0	开门	KA1	Q0. 0
停止按钮	SB2	I0. 1	关门	KA2	Q0. 1
行人通过	SB3	I0. 2			
门开限位开关	SQ1	I0. 3			
门关限位开关	SQ2	I0. 4			

2）系统的接线图。系统的接线图如图 4-41 所示。

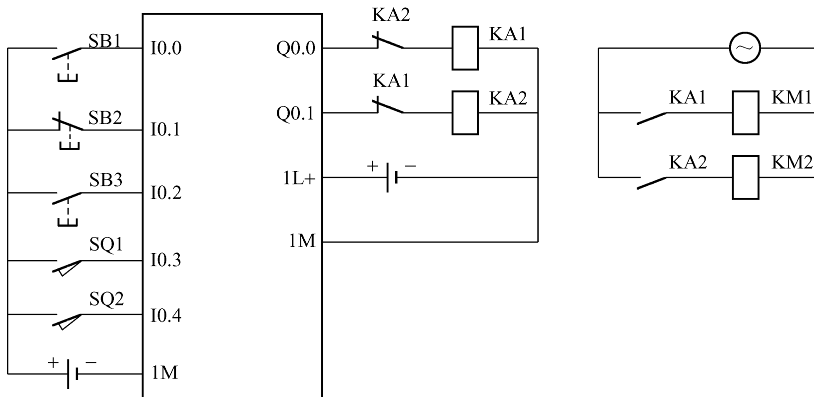


图 4-41　接线图

3. 编写程序

梯形图如图 4-42 所示。

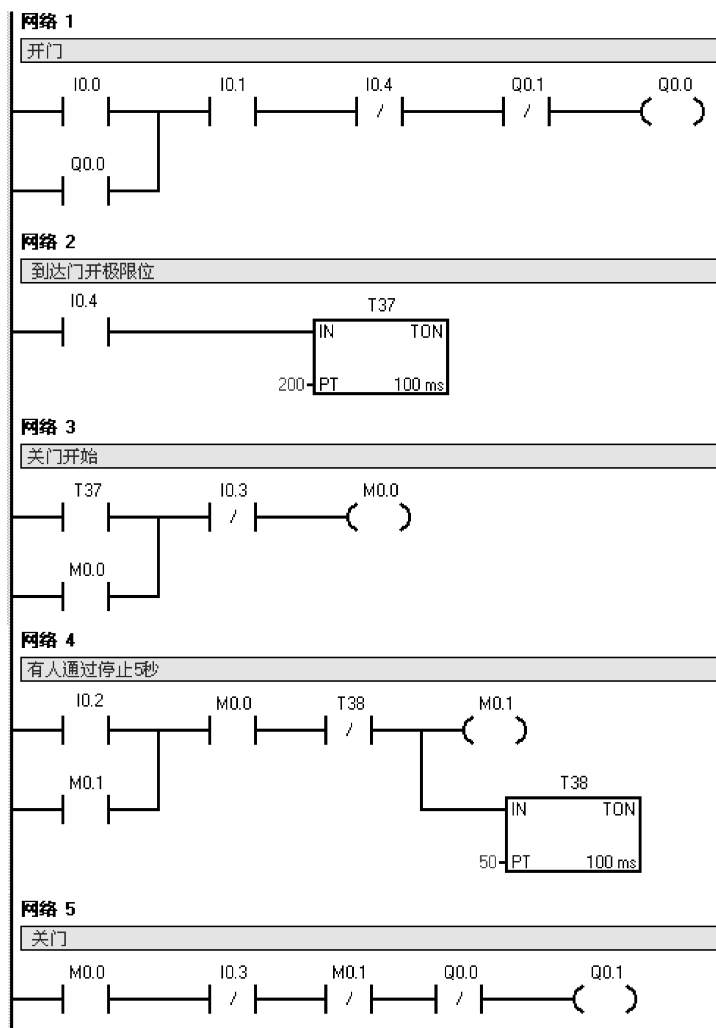


图 4-42　梯形图

4.3.2　计数器指令

计数器利用输入脉冲上升沿累计脉冲个数，S7-200 PLC 有递增计数（CTU）、增/减计数（CTUD）、递减计数（CTD）共三类计数指令。有的资料上将“增计数器”称为“加计数器”。计数器的使用方法和基本结构与定时器基本相同，主要由预置值寄存器、当前值寄存器和状态位等组成。

在梯形图指令符号中，CU 表示增 1 计数脉冲输入端，CD 表示减 1 计数脉冲输入端，R 表示复位脉冲输入端，LD 表示减计数器复位脉冲输入端，PV 表示预置值输入端，数据类型为 INT，预置值最大为 32 767。计数器的范围为 C0 ~ C255。

下面分别叙述 CTU、CTUD、CTD 三种类型计数器的使用方法。

1. 增计数器（CTU）

当 CU 端输入上升沿脉冲时，计数器的当前值增 1，当前值保存在 Cxxx（如 C0）中。当前值大于或等于预置值（PV）时，计数器状态位置 1。复位输入（R）有效时，计数器状态位复位，当前计数器值清 0。当计数值达到最大（32 767）时，计数器停止计数。增计数器指令和参数见表 4-12。

表 4-12　增计数器指令和参数

LAD	参　数	数 据 类 型	说　明	存　储　区
Cxxx CU　CTU R PV - PV	C xxx	常数	要启动的计数器号	C0 ~ C255
	CU	BOOL	加计数输入	I，Q，M，SM，T，C，V，S，L
	R	BOOL	复位	
	PV	INT	预置值	V，I，Q，M，SM，L，AI，AC，T，C，常数，＊VD，＊AC，＊LD，S

【例 4-20】 已知梯形图如图 4-43 所示，I0.0 和 I0.1 的时序如图 4-44 所示，请画出 Q0.0 的时序图。

【解】

CTU 为增计数器，当 I0.0 闭合 2 次时，常开触点 C0 闭合，Q0.0 输出为高电平“1”。当 I0.1 闭合时，计数器 C0 复位，Q0.0 输出为低电平“0”。

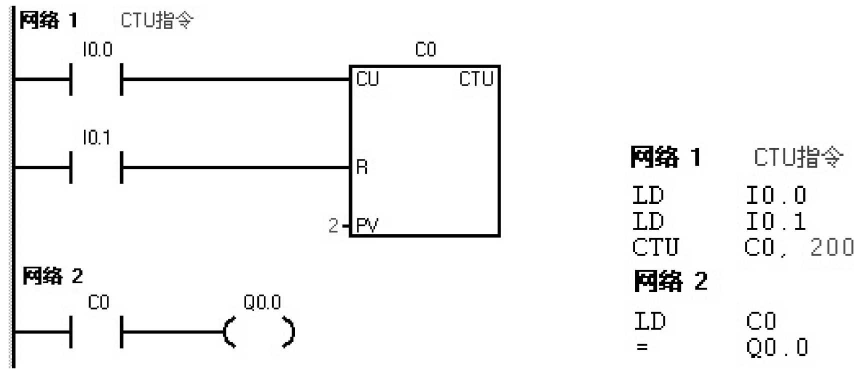


图 4-43　增计数器指令举例

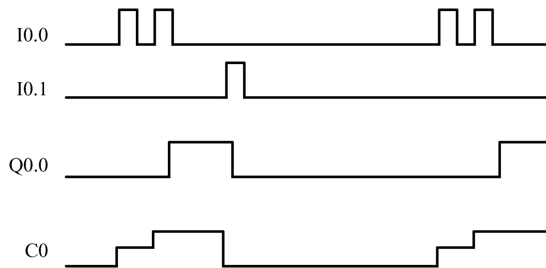


图 4-44　增计数器指令举例时序图

【例 4-21】设计一个程序，实现用一个单按钮控制一盏灯的亮和灭，即按奇数次按钮时，灯亮，按偶数次按钮时，灯灭。

【解】

当 I0.0 第一次闭合时，V0.0 接通一个扫描周期，使得 Q0.0 线圈得电一个扫描周期，当下一次扫描周期到达，Q0.0 常开触点闭合自锁，灯亮。

当 I0.0 第二次闭合时，V0.0 接通一个扫描周期，C0 计数为 2，Q0.0 线圈断电，使得灯灭，同时计数器复位。梯形图如图 4-45 所示。

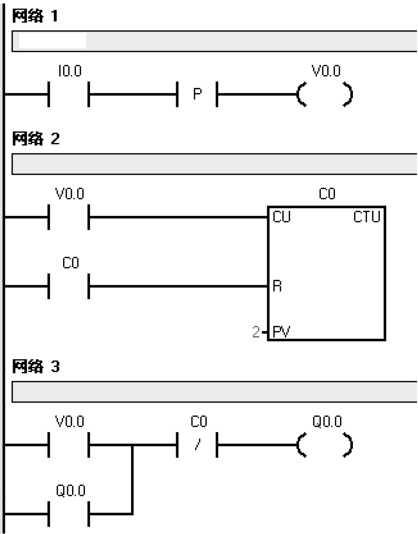


图 4-45　梯形图

【例 4-22】请编写一段程序，实现延时 6 h 后，点亮一盏灯，要求有起停控制。

【解】

S7-200 系列 PLC 的定时器的最大定时时间是 3 276.7 s，还不到 1 h，因此要延时 6 h 需要特殊处理，具体方法是用一个定时器 T37 定时 30 min，每次定时 30 min，计数器计数增加 1，直到计数 12 次，定时时间就是 6 h。梯形图如图 4-46 所示。

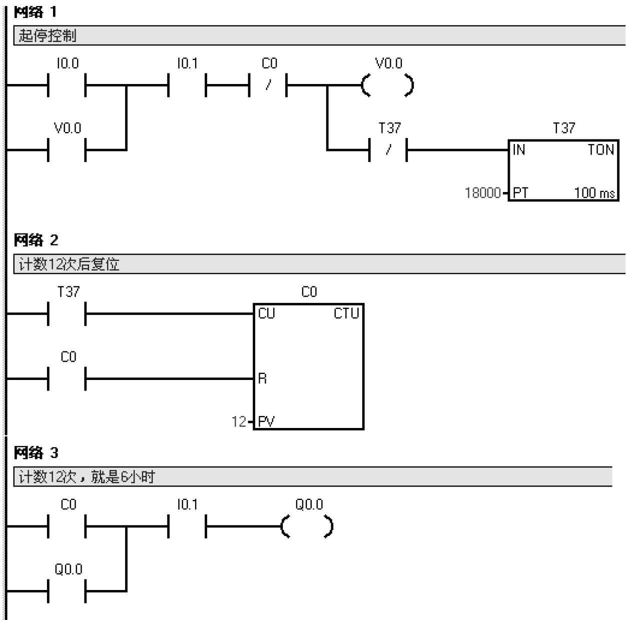


图 4-46　梯形图

【例 4-23】密码锁控制系统有 5 个按钮 SB1～SB5，其控制要求如下：

① SB1 为开锁按钮，按下 SB1 按钮，才可以开锁。

② SB2、SB3 为可按压按钮，开锁条件是：SB2 压 3 次，SB3 压 2 次。同时按下 SB2、SB3 时有顺序要求，先压 SB2，后压 SB3。

③ SB5 为不可按压的按钮，一旦按压，则系统报警。

④ SB4 为复位按钮，按压 SB4 后，可重新进行开锁作业，所有计数器被清 0。

【解】

程序如图 4-47 所示。

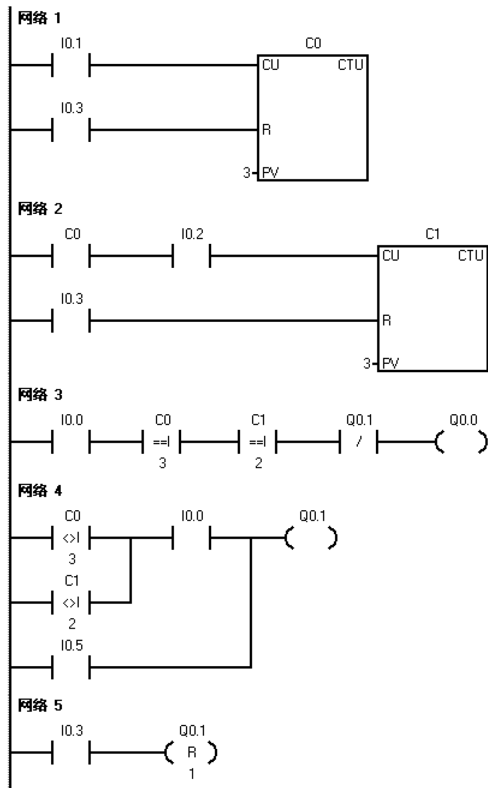


图 4-47　程序

2. 增/减计数器（CTUD）

增/减计数器有两个脉冲输入端。其中，CU 用于递增计数，CD 用于递减计数。执行增/减计数指令时，CU/CD 端的计数脉冲上升沿进行增 1/减 1 计数。当前值大于或等于计数器的预置值时，计数器状态位置位。复位输入（R）有效时，计数器状态位复位，当前值清 0。增/减计数器指令和参数见表 4-13。

表 4-13　增/减计数器指令和参数

LAD	参　　数	数 据 类 型	说　　明	存　储　区
Cxxx CU　CTUD CD R PV　PV	C xxx	常数	要启动的计数器号	C0 ~ C255
	CU	BOOL	加计数输入	I，Q，M，SM，T，C，V，S，L
	CD	BOOL	减计数输入	
	R	BOOL	复位	
	PV	INT	预置值	V，I，Q，M，SM，LW，AI，AC，T，C，常数，＊VD，＊AC，＊LD，S

【例 4-24】已知梯形图以及 I0.0、I0.1 和 I0.2 的时序如图 4-48 所示，请画出 Q0.0 的时序图。

【解】

利用增/减计数器输入端的通断情况分析 Q0.0 的状态。当 I0.0 接通 4 次时（4 个上升沿），C48 的常开触点闭合，Q0.0 通电；当 I0.0 接通 5 次时，C48 的计数为 5；接着当 I0.1 接通 2 次，此时 C48 的计数为 3，C48 的常开触点断开，Q0.0 断电；接着当 I0.0 接通 2 次，此时 C48 的计数为 5，C48 的计数大于或等于 4 时，C48 的常开触点闭合，Q0.0 通电；当 I0.2 接通时计数器复位，C48 的计数等于 0，C48 的常开触点断开，Q0.0 断电。

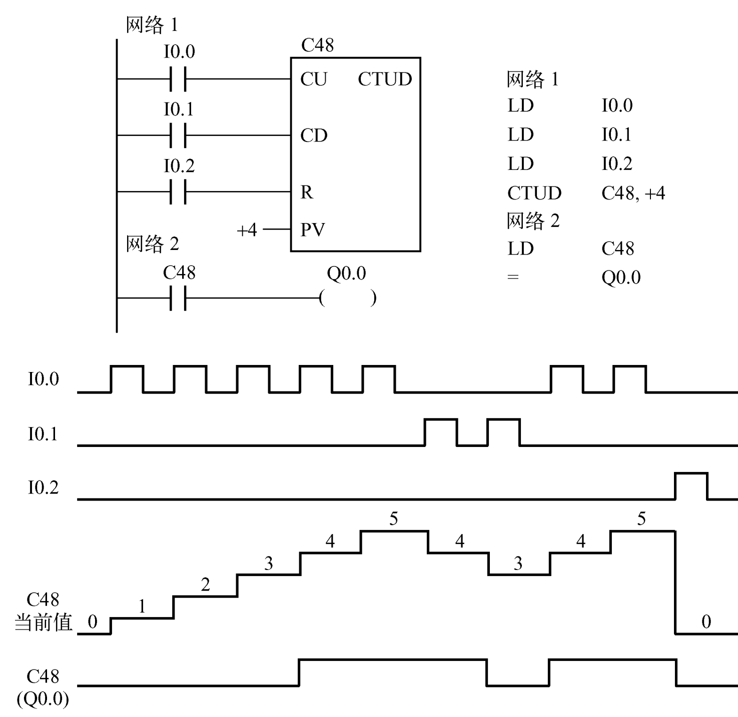


图 4-48　增/减计数器应用举例

【例 4-25】对某一端子上输入的信号进行计数。当计数达到某个变量存储器的设定值 10 时，PLC 控制灯泡发光，同时对该端子的信号进行减计数，当计数值小于另外一个变量存储器的设定值 5 时，PLC 控制灯泡熄灭，同时计数值清 0。请编写以上程序。

【解】

梯形图如图 4-49 所示。

3. 减计数器（CTD）

复位输入（LD）有效时，计数器把预置值（PV）装入当前值寄存器，计数器状态位复位。在 CD 端的每个输入脉冲上升沿，减计数器的当前值从预置值开始递减计数，当前值等于 0 时，计数器状态位置位，并停止计数。减计数器指令和参数见表 4-14。

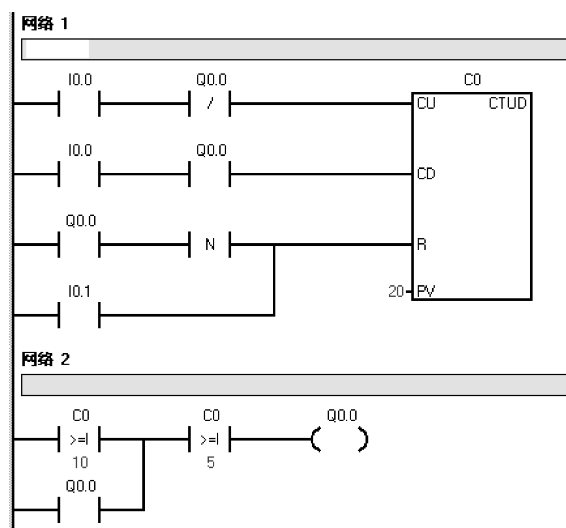


图 4-49 梯形图

表 4-14 减计数器指令和参数

LAD	参　数	数 据 类 型	说　明	存　储　区
Cxxx CD　CTD LD PV—PV	Cxxx	常数	要启动的计数器号	C0 ~ C255
	CD	BOOL	减计数输入	I，Q，M，SM，T，C，V，S，L
	LD	BOOL	预置值（PV）载入当前值	
	PV	INT	预置值	V，I，Q，M，SM，L，AI，AC，T，C，常数，*VD，*AC，*LD，S

【**例 4-26**】已知梯形图以及 I1.0 和 I2.0 的时序如图 4-50 所示，请画出 Q0.0 的时序图。

【**解**】

利用减计数器输入端的通断情况，分析 Q0.0 的状态。当 I2.0 接通时，计数器状态位复位，预置值3 被载入当前值寄存器；当 I1.0 接通 3 次时，当前值等于0，Q0.0 上电；当前值等于0 时，尽管 I1.0 接通，当前值仍然等于0。当 I2.0 接通期间，I1.0 接通，当前值不变。

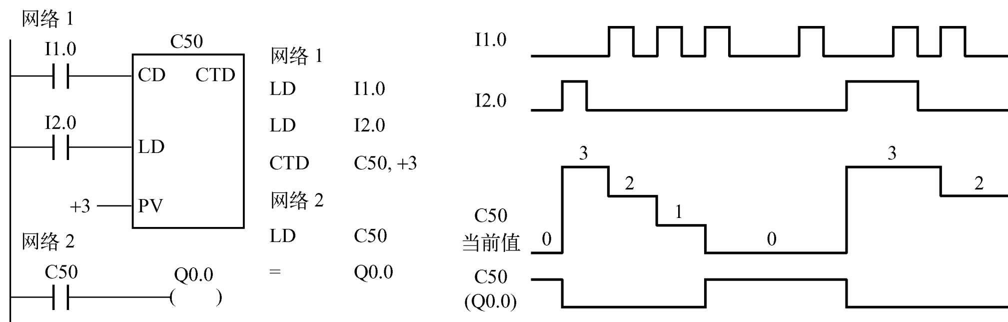


图 4-50 减计数器应用举例

4.3.3 基本指令的应用实例

【例 4-27】用 PLC 来控制电动机的正转、停止、反转（3 个按钮），不可以在运行当中改变方向。无论是正转还是反转，都必须用Y-△起动，从Y到△的延时时间为3 s，Y和△绝对不能同时导通。

【解】

1. PLC 的 I/O 分配

PLC 的 I/O 分配见表 4-15。

表 4-15 PLC 的 I/O 分配表

输入			输出		
名称	符号	输入点	名称	符号	输出点
正转按钮	SB1	I0.0	正转	KA1	Q0.0
反转按钮	SB2	I0.1	反转	KA2	Q0.1
停止按钮	SB3	I0.2	星形起动	KA3	Q0.2
			三角形运行	KA4	Q0.3

2. 系统的接线图

系统的接线图如图 4-51 所示。

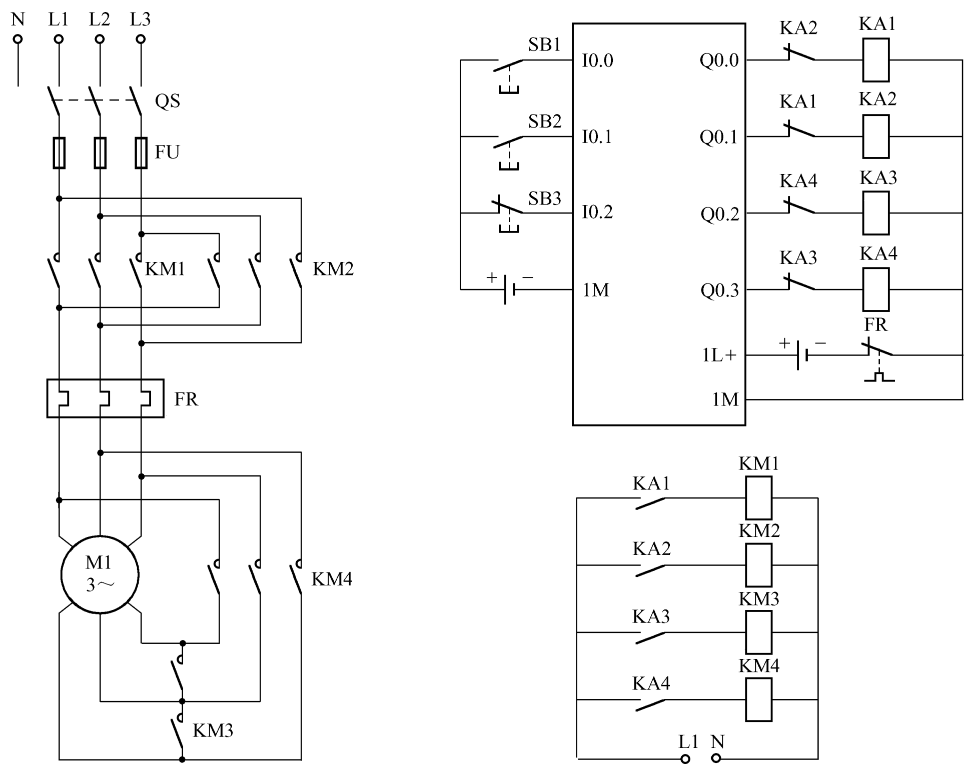


图 4-51 系统接线图

3. 编写程序

梯形图如图 4-52 所示。

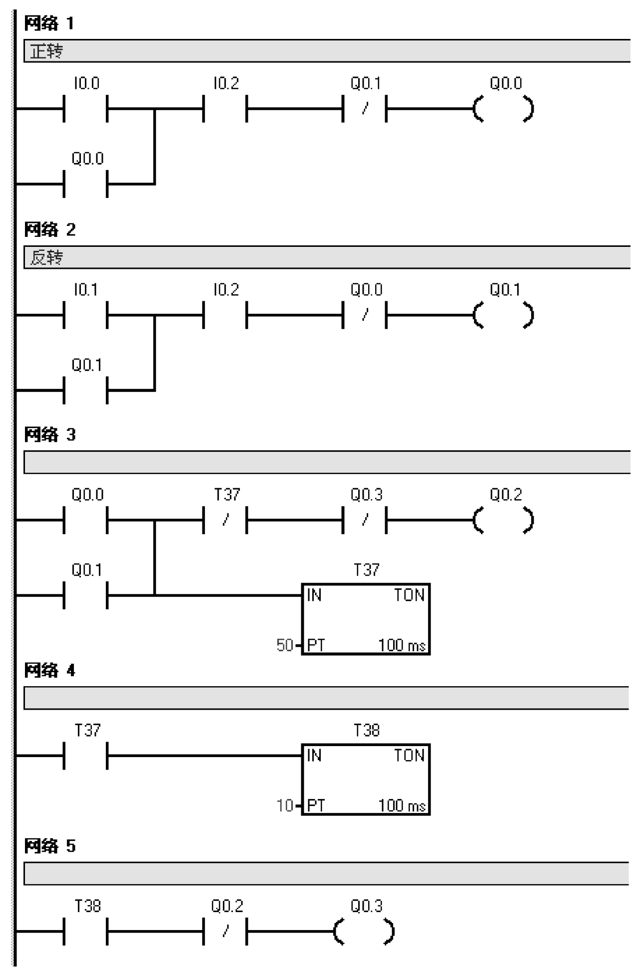


图 4-52　梯形图

【例 4-28】某十字路口的交通灯，如图 4-53 所示。其中，R、Y、G 分别代表红、黄、绿的交通灯。要完成如下功能：

① 设置起动、停止按钮。正常起动情况下，东西向绿灯亮 30 s，转东西向绿灯以 0.5 s 闪烁 4 s，转东西向黄灯亮 3 s，转南北向绿灯亮 30 s，转南北向绿灯以 0.5 s 闪烁 4 s，转南北向黄灯亮 3 s，再转东西向绿灯亮 30 s，以此类推。

② 在东西向绿灯亮时，南北向应显示红灯。同理，南北向绿灯亮时，东西向应显示红灯。

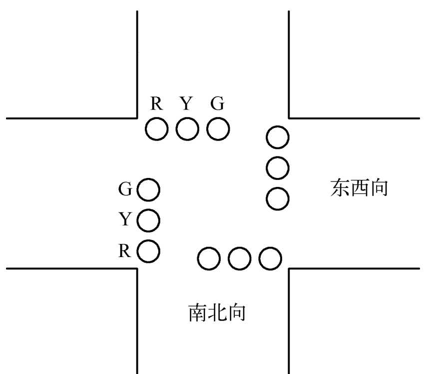


图 4-53　交通灯示意图

【解】

1. 绘制时序图

由于十字交通灯的逻辑比较复杂，为了方便编写程序，可先根据题意绘制时序图，如图 4-54 所示。

把不同颜色的灯的亮灭情况，罗列出来，具体如下：

（1）东西方向

$T<30$ s，绿灯亮；30 s≤$T<34$ s，绿灯闪烁；34 s≤$T<37$ s，黄灯亮；37 s≥T≥74 s，红灯亮。

（2）南北方向

$T<37$ s，红灯亮；37 s≤$T<67$ s，绿灯亮；67 s≤$T<71$ s，绿灯闪烁；74 s≥T≥71 s，黄灯亮。

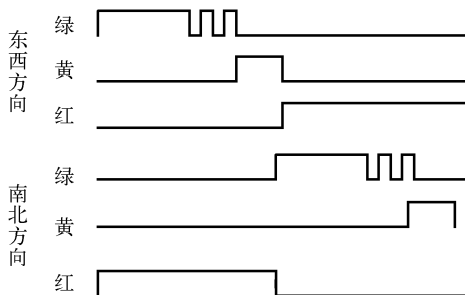


图 4-54　交通灯时序图

2. PLC 的 I/O 分配

PLC 的 I/O 分配见表 4-16。

表 4-16　PLC 的 I/O 分配表

输　　入			输　　出		
名　　称	符　　号	输　入　点	名　　称	符　　号	输　出　点
开始按钮	SB1	I0. 0	绿灯（东西）	HL1	Q0. 0
停止按钮	SB2	I0. 1	黄灯（东西）	HL2	Q0. 1
			红灯（东西）	HL3	Q0. 2
			绿灯（南北）	HL4	Q0. 3
			黄灯（南北）	HL5	Q0. 4
			红灯（南北）	HL6	Q0. 5

3. 控制系统的接线

交通灯控制系统的接线比较简单，如图 4-55 所示。

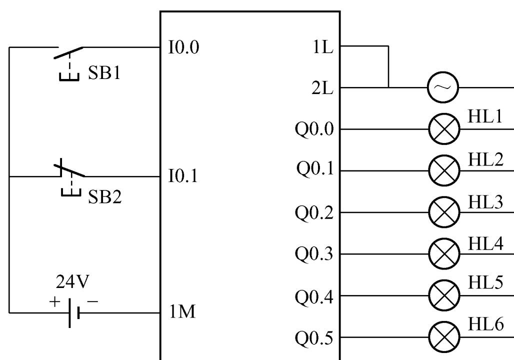


图 4-55　系统接线图

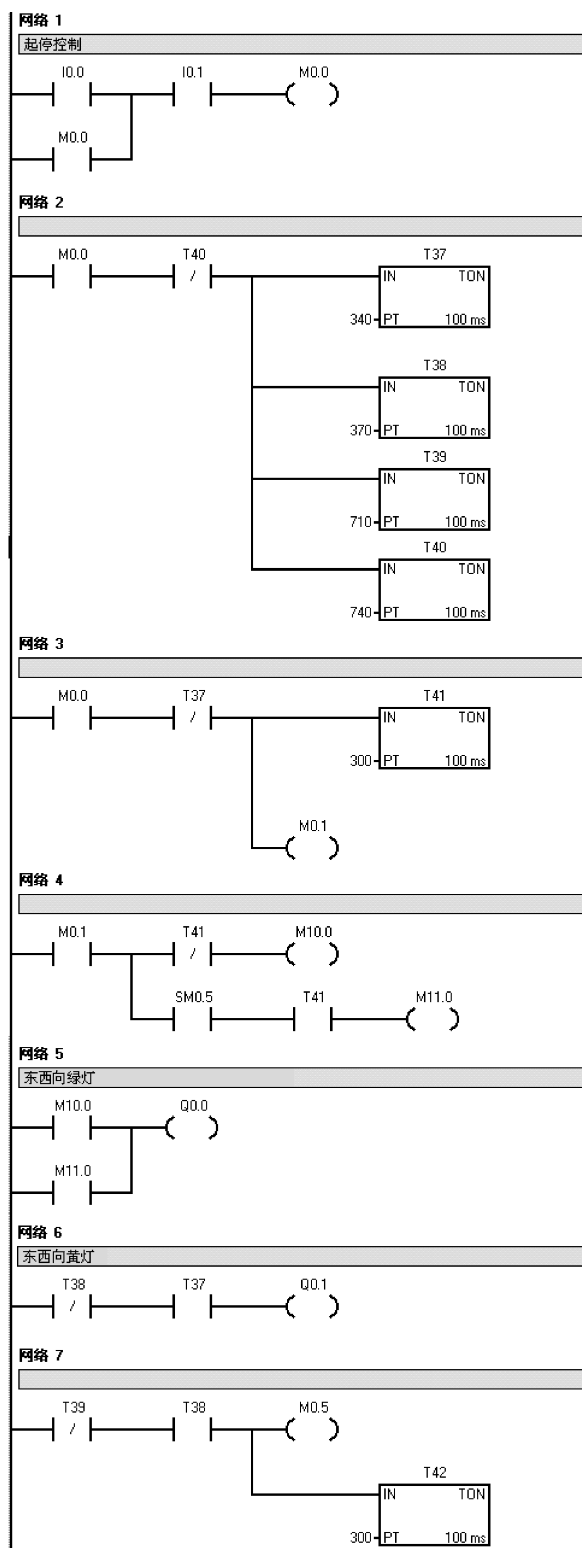


图 4-56　交通灯梯形图（基本指令）

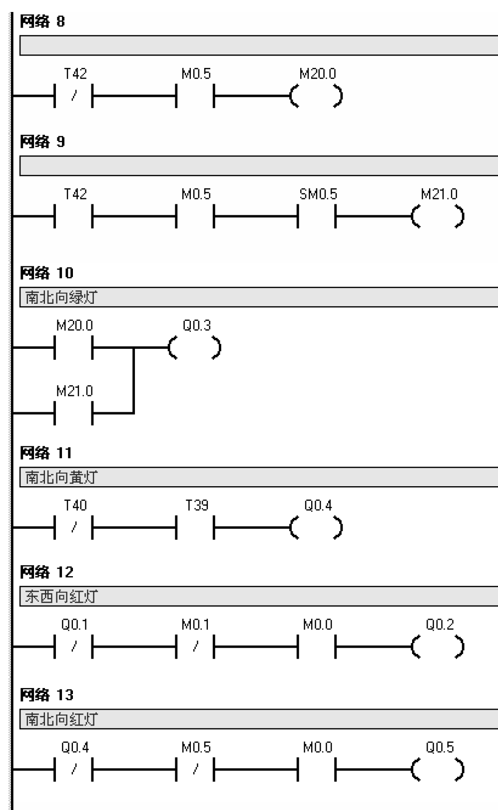


图 4-56　交通灯梯形图（基本指令）（续）

【例 4-29】 现有一套三级输送机，用于实现货料的传输，每一级输送机由一台交流电动机进行控制，电动机为 M1、M2、M3，分别由接触器 KM1、KM2、KM3、KM4、KM5、KM6 控制电动机的正反转运行。

系统的结构示意图如图 4-57 所示。

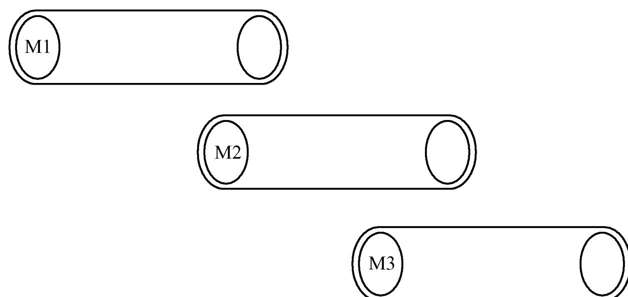


图 4-57　系统的结构示意图

【解】

1）控制任务描述

① 当装置上电时，系统进行复位，所有电动机停止运行。

② 当手/自动转换开关 SA1 打到左边时系统进入自动状态。按下系统起动按钮 SB1 时，电动机 M3 首先正转起动，运转 5 s 以后，电动机 M2 正转起动，当电动机 M2 运转 6 s 以后，电动机 M1 正转起动，此时系统完成起动过程，进入正常运转状态。

③ 当按下系统停止按钮 SB2 时，电动机 Ml 首先停止，当电动机 Ml 停止 4 s 以后，电动机 M2 停止，当 M2 停止 4 s 以后，电动机 M3 停止。系统在起动过程中按下停止按钮 SB2，电动机按起动的顺序反向停止运行。

④ 当系统按下急停按钮 SB9 时，两台电动机要求停止工作，直到急停按钮取消时，系统恢复到之前状态。

⑤ 当手/自动转换开关 SA1 打到右边时系统进入手动状态，系统只能由手动开关控制电动机的运行。通过手动开关（SB3 ~ SB8），操作者能控制两台电动机的正反转运行，实现货物的手动传输。

2）编写程序。根据系统的功能要求，完成下列操作。

根据系统要求，将开关量输入/输出地址在电气原理接线图上进行标识并将 PLC 的电气原理接线图补充完整。

电气原理图如图 4-58 所示，梯形图如图 4-59 所示。

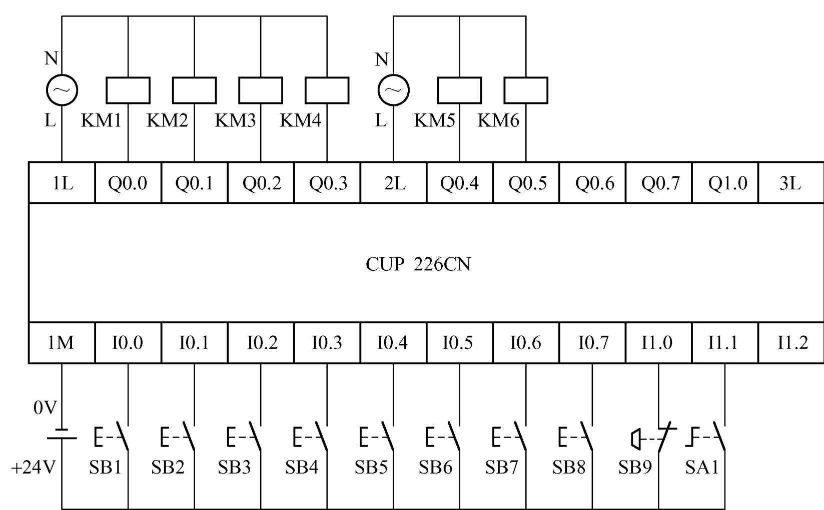


图 4-58 电气原理图

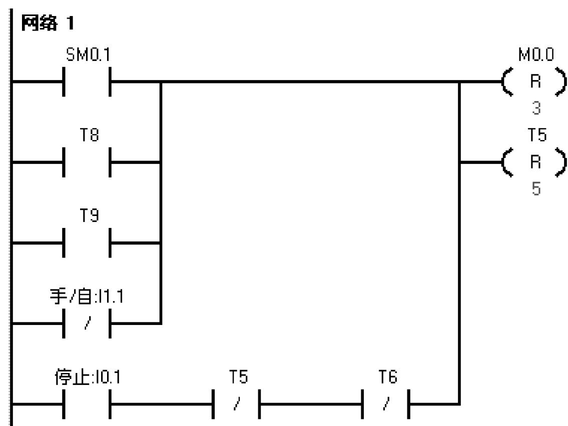
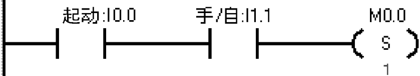


图 4-59 梯形图

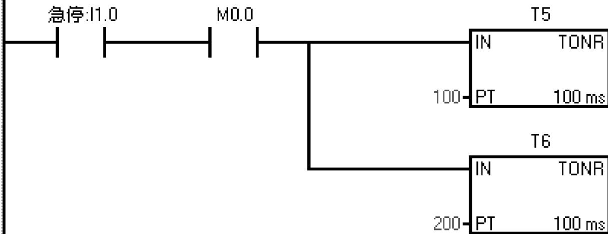
符号	地址	注释
手/自	I1.1	手/自
停止	I0.1	停止

网络 2



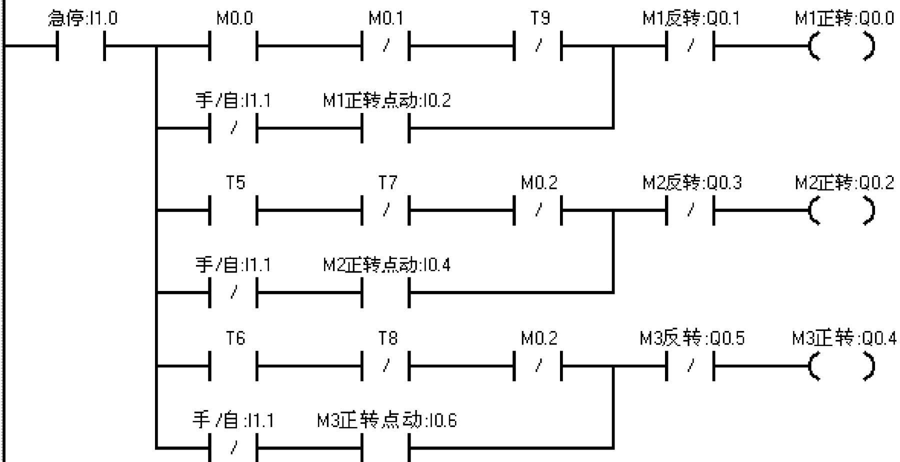
符号	地址	注释
起动	I0.0	
手/自	I1.1	手/自

网络 3



符号	地址	注释
急停	I1.0	急停

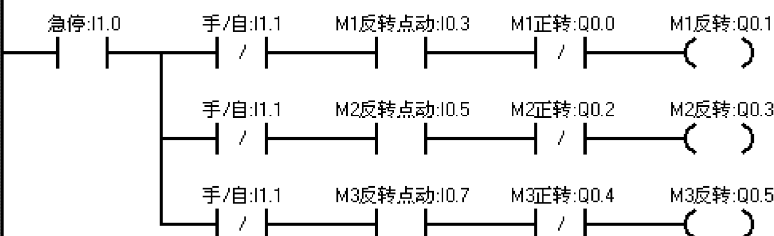
网络 4



符号	地址	注释
M1反转	Q0.1	M1反转
M1正转	Q0.0	M1正转
M1正转点动	I0.2	M1正转点动
M2反转	Q0.3	M2反转
M2正转	Q0.2	M2正转
M2正转点动	I0.4	M2正转点动
M3反转	Q0.5	M3反转
M3正转	Q0.4	M3正转
M3正转点动	I0.6	M3正转点动
急停	I1.0	急停
手/自	I1.1	手/自

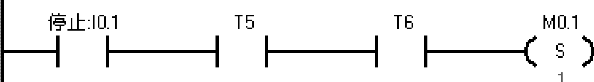
图 4-59　梯形图（续）

网络 5



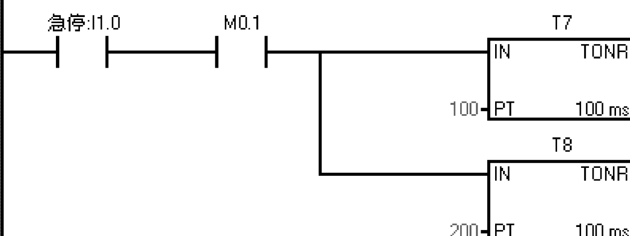
符号	地址	注释
M1反转	Q0.1	M1反转
M1反转点动	I0.3	M1反转点动
M1正转	Q0.0	M1正转
M2反转	Q0.3	M2反转
M2反转点动	I0.5	M2反转点动
M2正转	Q0.2	M2正转
M3反转	Q0.5	M3反转
M3反转点动	I0.7	M3反转点动
M3正转	Q0.4	M3正转
急停	I1.0	急停
手/自	I1.1	手/自

网络 6



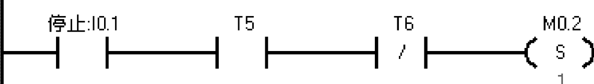
符号	地址	注释
停止	I0.1	停止

网络 7



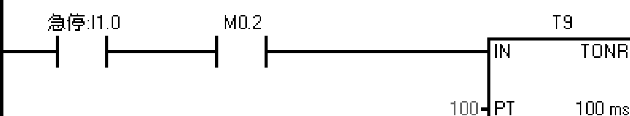
符号	地址	注释
急停	I1.0	急停

网络 8



符号	地址	注释
停止	I0.1	停止

网络 9



符号	地址	注释
急停	I1.0	急停

图 4-59　梯形图（续）

【**例 4-30**】如图 4-60 所示的气动机械手由 3 个气缸组成，即气缸 A、B、C。其接线图如图 4-61 所示。其工作过程是：当接近开关 SQ0 检测到有物体时，系统开始工作，气缸 A 向左运行；到极限位置 SQ2 后，气缸 B 向下运行，直到极限位置 SQ4 为止；接着手指气缸 C 抓住物体，延时 1 s；然后气缸 B 向上运行；到极限位置 SQ3 后，气缸 A 向右运行；到极限位置 SQ1，此时手指气缸 C 释放物体，并延时 1 s，完成搬运工作。电磁阀 YV1 上电气缸 A 向左运行，电磁阀 YV2 上电气缸 A 向右运行，电磁阀 YV3 上电，气缸 B 向下运行，电磁阀 YV4 上电，气缸 B 向上运行，电磁阀 YV5 上电，气缸 C 夹紧，电磁阀 YV5 断电，气缸 C 松开。请画出接线图、流程图和梯形图。

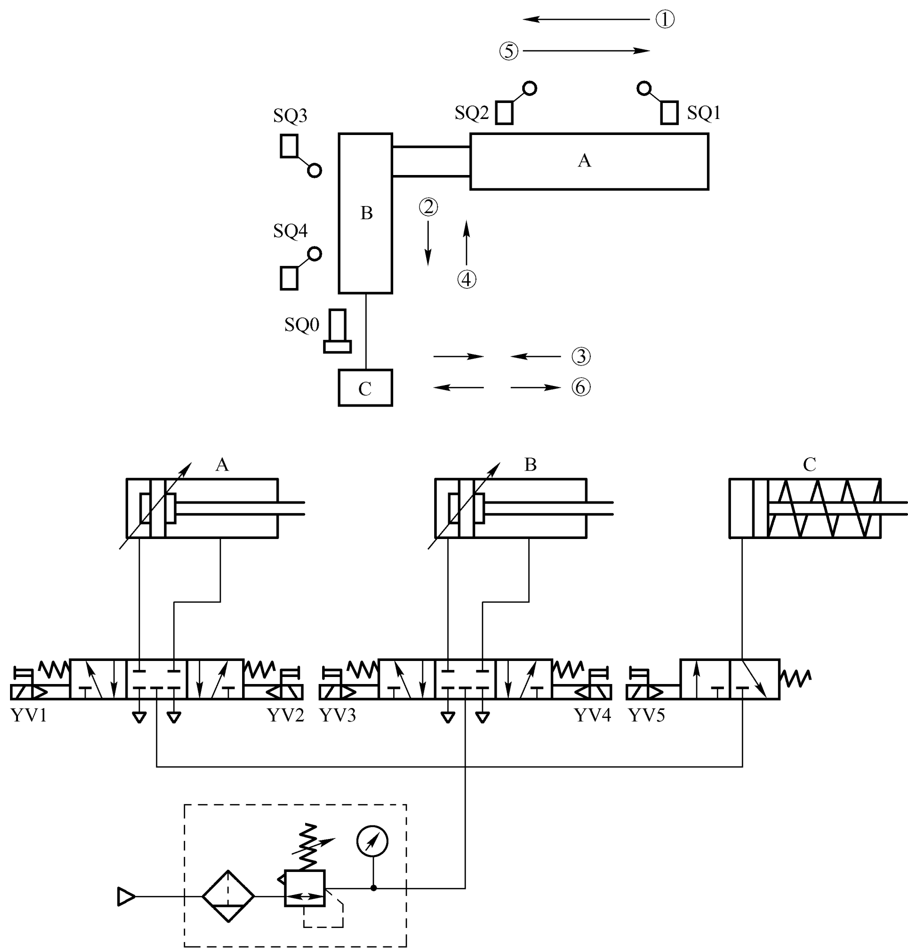


图 4-60　机械手示意图及气动原理图

【**解**】

这个运动逻辑看起来比较复杂，如果不掌握规律，则很难设计出正确的梯形图，一般先根据题意画出流程图，再根据流程图写出布尔表达式，如图 4-62 所示。布尔表达式是有规律的，当前步的步名对应的继电器（如 M0.1）等于上一步的步名对应的继电器（M0.0）与上一步的转换条件（I0.2）的乘积，再加上当前步的步名对应的继电器（M0.1）与下一步的步名对应的继电器非的乘积（$\overline{\text{M0.2}}$），其他的布尔表达式的写法类似，最后根据布尔表

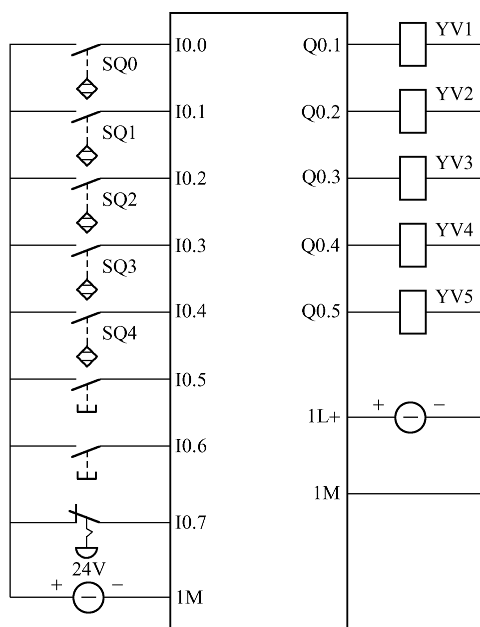


图 4-61　机械手接线图

达式画出梯形图，如图 4-63 所示。在整个过程中，流程图是关键，也是难点，而根据流程图写出布尔表达式和画出梯形图比较简单。读者可在学完 5.1 节后再看图 4-62 的流程图。其中 I0.5 是复位按钮，I0.6 是起动按钮，I0.7 是急停按钮。

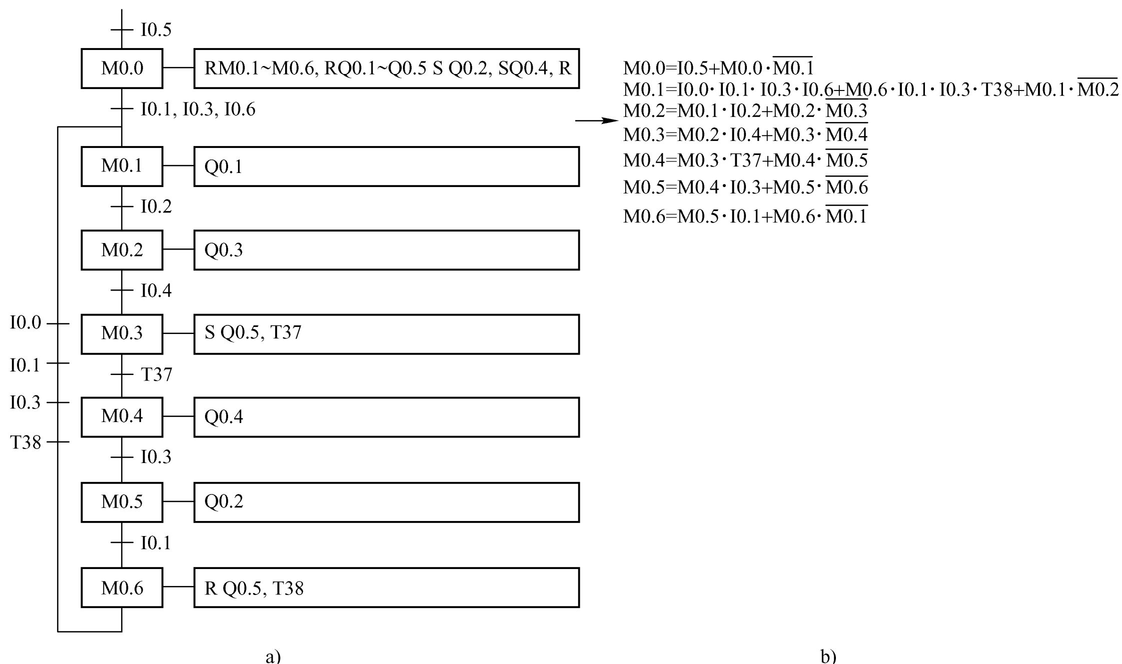


图 4-62　机械手的流程图和布尔表达式对应关系图

a）流程图　b）布尔表达式

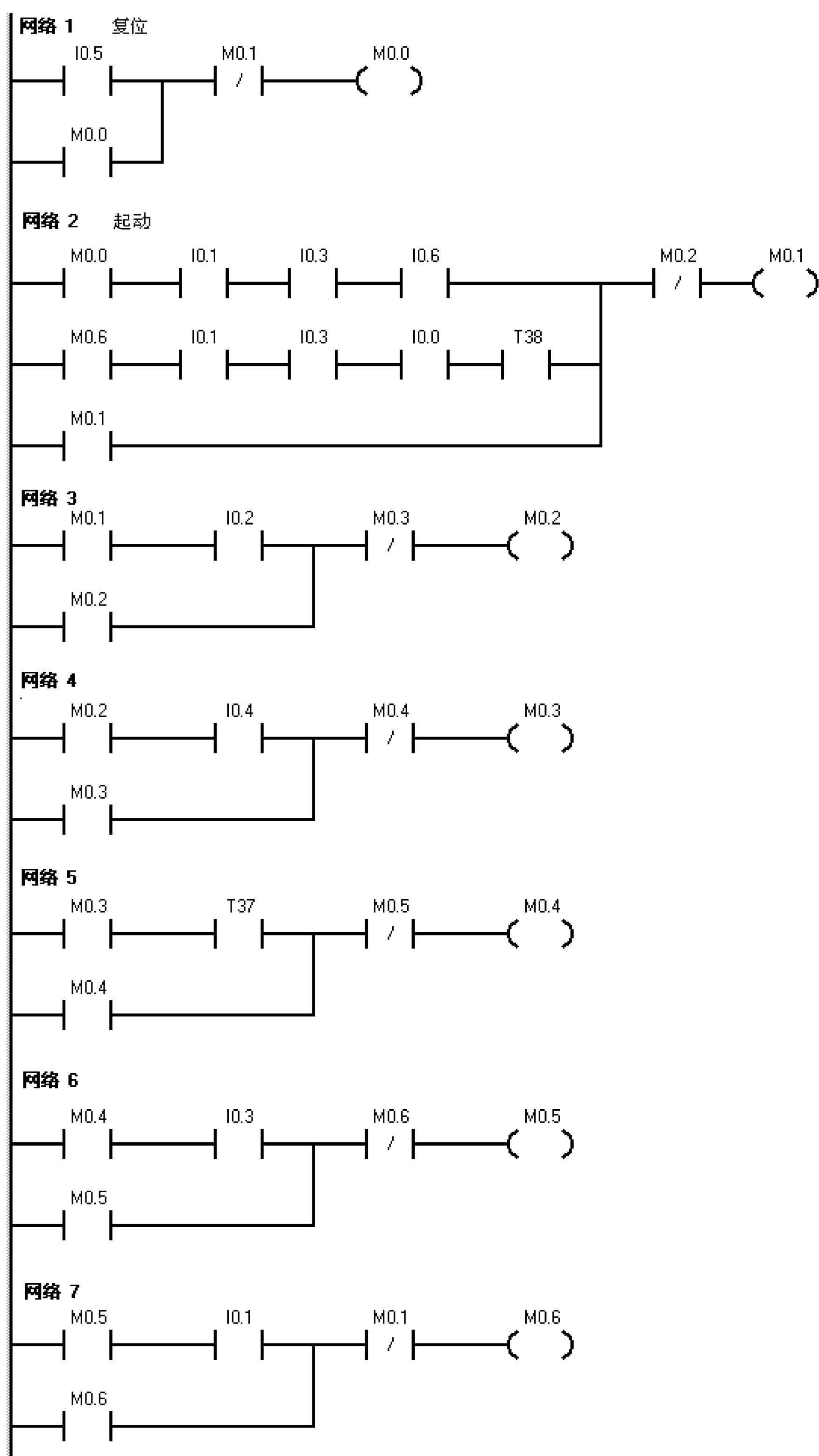


图 4-63　机械手的梯形图

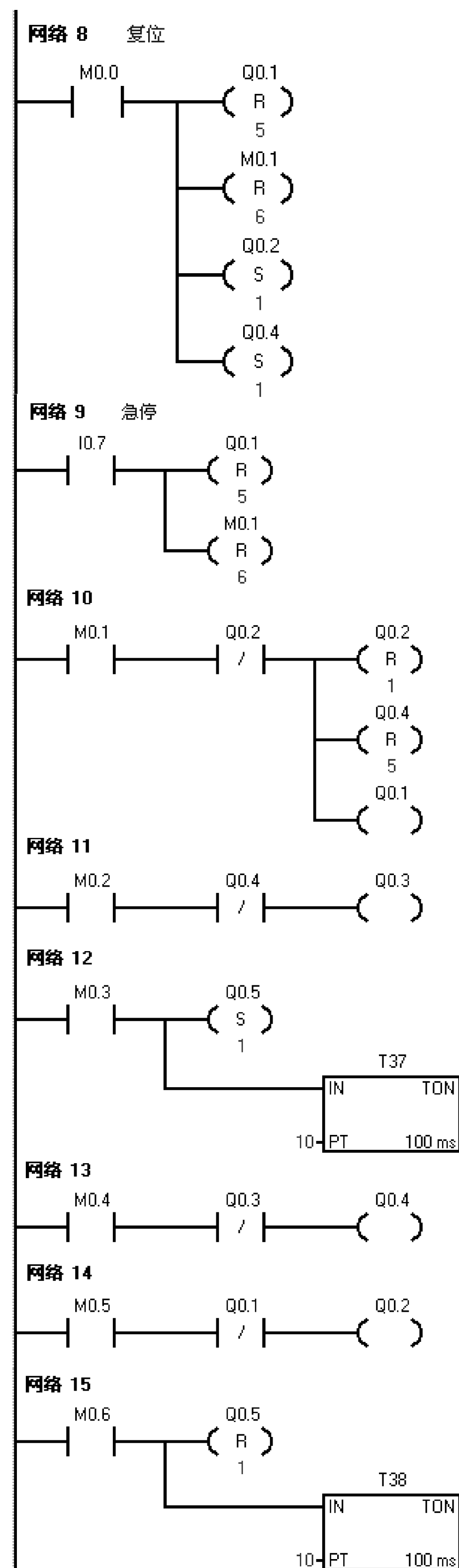


图 4-63 机械手的梯形图（续）

【关键点】复位和急停的结果是不同的，当按下复位按钮时，气缸回到初始位置，这点很重要，但按下急停按钮后，气缸 A 和 B 停止在当前位置，只有气缸 C 松开。因为气缸 C 是单作用气缸，会自动复位。

4.4 功能指令

为了满足用户的一些特殊要求，20 世纪 80 年代开始，众多 PLC 制造商就在小型机上加入了功能指令（或称应用指令）。这些功能指令的出现，大大拓宽了 PLC 的应用范围。S7－200 系列 PLC 的功能指令极其丰富，主要包括算术运算、数据处理、逻辑运算、高速处理、PID、中断、实时时钟和通信指令。PLC 在处理模拟量时，一般要进行数据处理。

4.4.1 比较指令

STEP 7 提供了丰富的比较指令，可以满足用户的多种需要。STEP 7 中的比较指令可以对下列数据类型的数值进行比较。

① 两个字节的比较（每个字节为 8 位）。

② 两个字符串的比较（每个字符串为 8 位）。

③ 两个整数的比较（每个整数为 16 位）。

④ 两个双整数的比较（每个双整数为 32 位）。

⑤ 两个实数的比较（每个实数为 32 位）。

【关键点】一个整数和一个双整数是不能直接进行比较的，因为它们之间的数据类型不同。一般先将整数转换成双整数，再对两个双整数进行比较。

比较指令有等于（EQ）、不等于（NQ）、大于（GT）、小于（LQ）、大于或等于（GE）和小于或等于（LE）。比较指令对输入 IN1 和 IN2 进行比较。

比较指令是将两个操作数按指定的条件作比较，比较条件满足时，触点闭合，否则断开。比较指令为上、下限控制等提供了极大的方便。在梯形图中，比较指令可以装入，也可以串、并联。

1. 等于比较指令

等于指令有字节等于比较指令、整数等于比较指令、双整数等于比较指令、符号等于比较指令和实数等于比较指令五种。整数等于比较指令和参数见表 4－17。

表 4－17　整数等于比较指令和参数

LAD	参　数	数据类型	说　明	存　储　区
IN1 ─┤==I├─ IN2	IN1	INT	比较的第一个数值	I，Q，M，S，SM，T，C，V，L，AI，AC，常数，＊VD，＊LD，＊AC
	IN2	INT	比较的第二个数值	

用一个例子来说明整数等于比较指令，梯形图和指令表如图 4－64 所示。当 I0.0 闭合时，激活比较指令，MW0 中的整数和 MW2 中的整数比较，若两者相等，则 Q0.0 输出为“1”，若两者不相等，则 Q0.0 输出为“0”。在 I0.0 不闭合时，Q0.0 的输出为“0”。IN1

和 IN2 可以为常数。

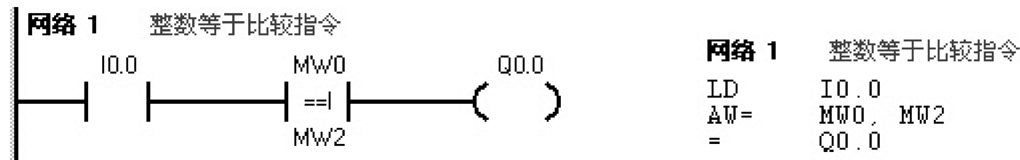


图 4-64 整数等于比较指令举例

图 4-64 中，若无常开触点 I0.0，则每次扫描时都要进行整数比较运算。

双整数等于比较指令和实数等于比较指令的使用方法与整数等于比较指令类似，只不过 IN1 和 IN2 的参数类型分别为双整数和实数。

2. 不等于比较指令

不等于比较指令有字节不等于比较指令、整数不等于比较指令、双整数不等于比较指令、符号不等于比较指令和实数不等于比较指令五种。整数不等于比较指令和参数见表 4-18。

表 4-18 整数不等于比较指令和参数

LAD	参数	数据类型	说明	存储区
IN1 ┤<>I├ IN2	IN1	INT	比较的第一个数值	I，Q，M，S，SM，T，C，V，L，AI，AC，常数，*VD，*LD，*AC
	IN2	INT	比较的第二个数值	

用一个例子来说明整数不等于比较指令，梯形图和指令表如图 4-65 所示。当 I0.0 闭合时，激活比较指令，MW0 中的整数和 MW2 中的整数比较，若两者不相等，则 Q0.0 输出为“1”，若两者相等，则 Q0.0 输出为“0”。在 I0.0 不闭合时，Q0.0 的输出为“0”。IN1 和 IN2 可以为常数。

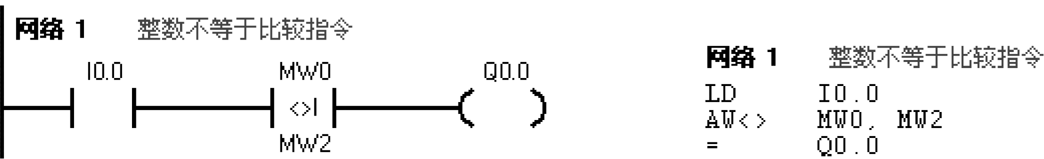


图 4-65 整数不等于比较指令举例

双整数不等于比较指令和实数不等于比较指令的使用方法与整数不等于比较指令类似，只不过 IN1 和 IN2 的参数类型分别为双整数和实数。使用比较指令的前提是数据类型必须相同。

3. 小于比较指令

小于比较指令有字节小于比较指令、整数小于比较指令、双整数小于比较指令和实数小于比较指令四种。双整数小于比较指令和参数见表 4-19。

表 4-19　双整数小于比较指令和参数

LAD	参　数	数据类型	说　明	存　储　区
IN1 ┤<D├ IN2	IN1	DINT	比较的第一个数值	I, Q, M, S, SM, V, L, HC, AC, 常数, ＊VD, ＊LD, ＊AC
	IN2	DINT	比较的第二个数值	

用一个例子来说明双整数小于比较指令，梯形图和指令表如图 4-66 所示。当 I0.0 闭合时，激活双整数小于比较指令，MD0 中的双整数和 MD4 中的双整数比较，若前者小于后者，则 Q0.0 输出为“1”，否则，Q0.0 输出为“0”。在 I0.0 不闭合时，Q0.0 的输出为“0”。IN1 和 IN2 可以为常数。

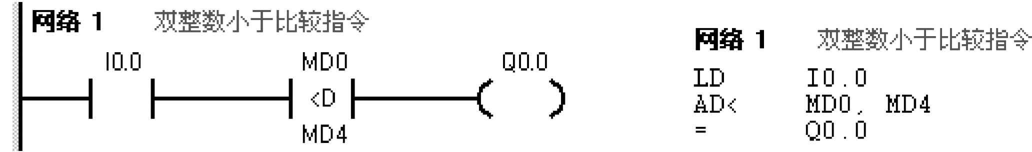


图 4-66　双整数小于比较指令举例

整数小于比较指令和实数小于比较指令的使用方法与双整数小于比较指令类似，只不过 IN1 和 IN2 的参数类型分别为整数和实数。使用比较指令的前提是数据类型必须相同。

4. 大于等于比较指令

大于等于比较指令有字节大于等于比较指令、整数大于等于比较指令、双整数大于等于比较指令和实数大于等于比较指令四种。实数大于等于比较指令和参数见表 4-20。

表 4-20　实数大于等于比较指令和参数

LAD	参　数	数据类型	说　明	存　储　区
IN1 ┤>=R├ IN2	IN1	REAL	比较的第一个数值	I, Q, M, S, SM, V, L, AC, 常数, ＊VD, ＊LD, ＊AC
	IN2	REAL	比较的第二个数值	

用一个例子来说明实数大于等于比较指令，梯形图和指令表如图 4-67 所示。当 I0.0 闭合时，激活比较指令，MD0 中的实数和 MD4 中的实数比较，若前者大于或者等于后者，则 Q0.0 输出为“1”，否则，Q0.0 输出为“0”。在 I0.0 不闭合时，Q0.0 的输出为“0”。IN1 和 IN2 可以为常数。

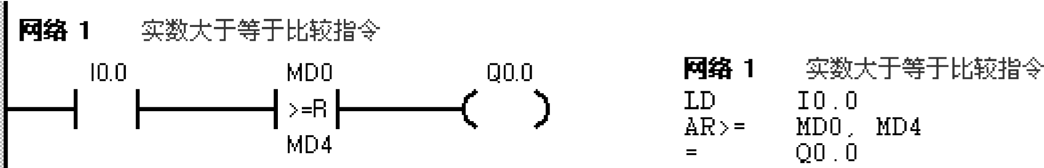


图 4-67　实数大于等于比较指令举例

整数大于等于比较指令和双整数大于等于比较指令的使用方法与实数大于等于比较指令类似，只不过 IN1 和 IN2 的参数类型分别为整数和双整数。使用比较指令的前提是数据类型必须相同。

小于等于比较指令和小于比较指令类似，大于比较指令和大于等于比较指令类似，在此不再讲述小于等于比较指令和大于比较指令。

【例 4-31】题目为例 4-28，用比较指令编写交通灯程序。

前面用基本指令编写了交通灯的程序，相对比较复杂，初学者不易掌握，但对照例 4-28 的时序图，用比较指令编写程序就非常容易了，程序如图 4-68 所示。

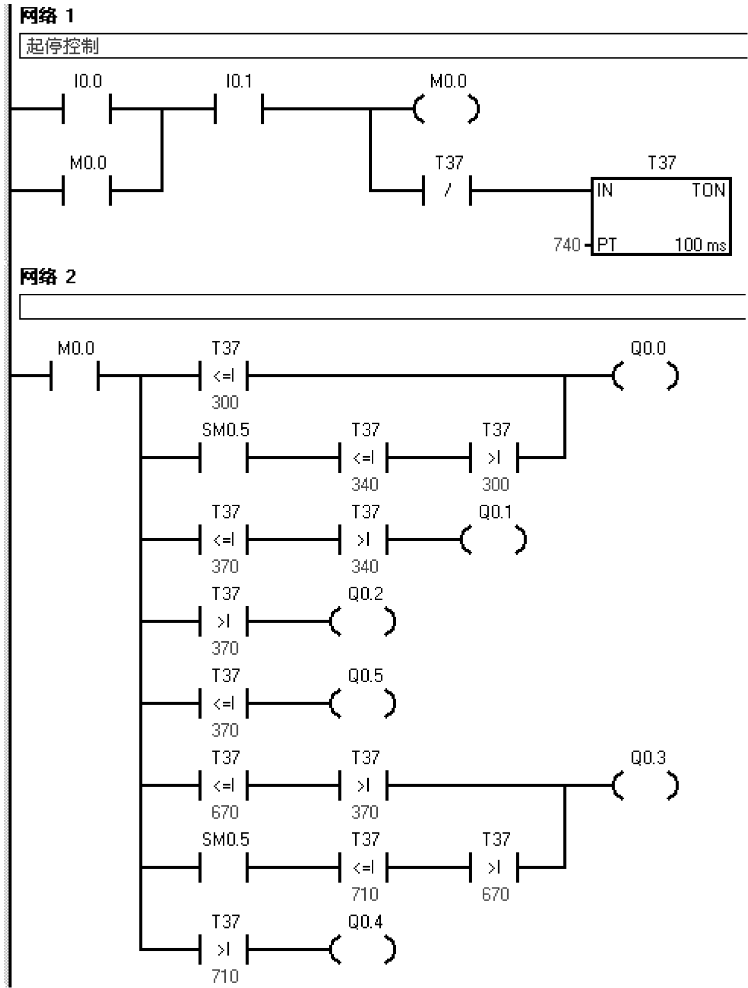


图 4-68　交通灯程序（比较指令）

4. 4. 2　数据处理指令

数据处理指令包括数据传送指令、交换/字节填充指令及移位指令等。数据传送指令非常有用，特别在数据初始化、数据运算和通信时经常用到。

1. 数据传送指令

数据传送指令有字节、字、双字和实数的单个数据传送指令，还有以字节、字、双字为单位的数据块传送指令，用以实现各存储器单元之间的数据传送和复制。

单个数据传送指令一次完成一个字节、字或双字的传送。以下仅以字节传送指令为例说明传送指令的使用方法，字节传送指令格式见表 4-21。

表 4-21　字节传送指令格式

LAD	参　数	数据类型	说　明	存　储　区
IN1 ==I IN2	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	OUT	BYTE	目的地地址	V，I，Q，M，S，SM，L，AC，＊VD，＊LD，＊AC，常数（OUT 中无常数）
	IN	BYTE	源数据	

当使能端输入 EN 有效时，将输入端 IN 中的字节传送至 OUT 指定的存储器单元输出。输出端 ENO 的状态和使能端 EN 的状态相同。

【例 4-32】VB0 中的数据为 20，程序如图 4-69 所示，试分析运行结果。

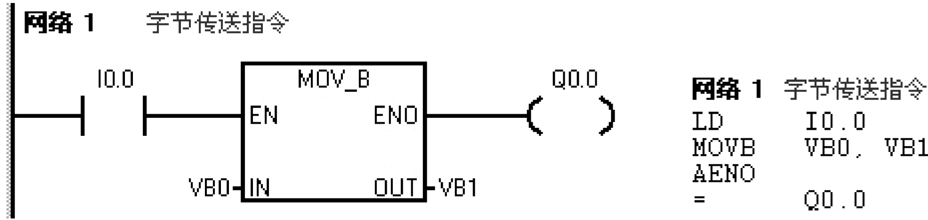


图 4-69　字节传送指令应用举例

【解】

当 I0.0 闭合时，执行字节传送指令，VB0 和 VB1 中的数据都为 20，同时 Q0.0 输出高电平；当 I0.0 闭合后断开，VB0 和 VB1 中的数据仍都为 20，但 Q0.0 输出低电平。

字、双字和实数传送指令的使用方法与字节传送指令类似，在此不再说明。

【关键点】读者若将输出 VB1 改成 VW1，则程序出错。因为字节传送的操作数不能为字。

【例 4-33】如图 4-70 所示的电动机Y-△起动的电气原理图，请编写程序。

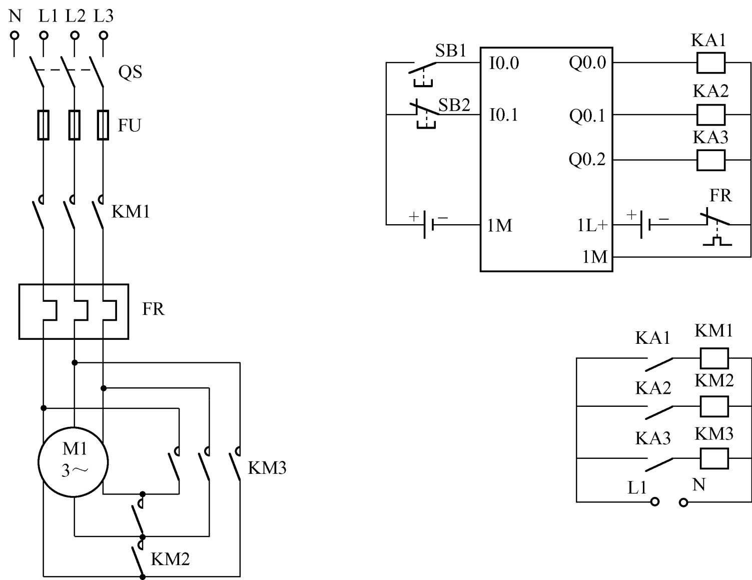


图 4-70　原理图

【解】

前 10 s，Q0.0 和 Q0.1 线圈得电，星形起动，从第 10～11 s 只有 Q0.0 得电，从 11 s 开始，Q0.0 和 Q0.2 线圈得电，电动机为三角形运行。程序如图 4-71 所示。这种方法编写程序很简单，但浪费了宝贵的输出点资源。

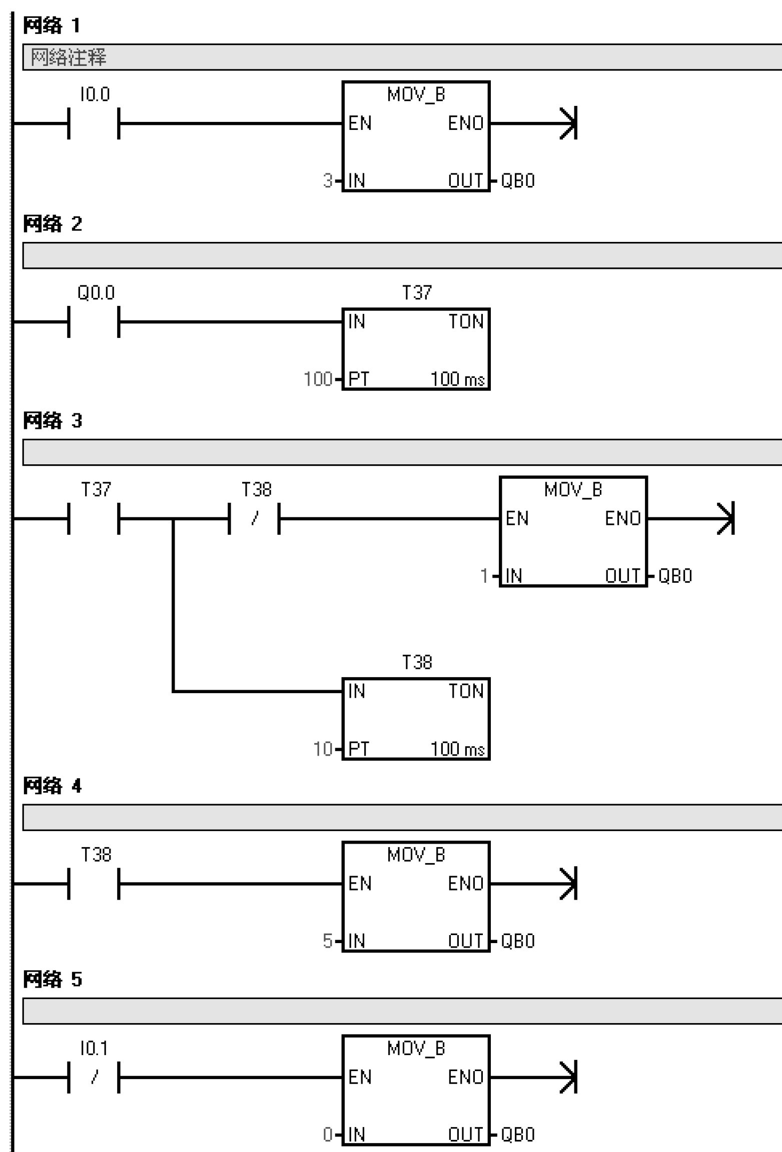


图 4-71　电动机Y－△起动程序

2. 数据块传送指令（BLKMOV）

数据块传送指令一次完成 N 个数据的成组传送，数据块传送指令是一个效率很高的指令，应用很方便。有时，使用一条数据块传送指令可以取代多条传送指令，其指令格式见表 4-22。

表 4-22　数据块的传送指令格式

LAD	参　　数	数据类型	说　　明	存　储　区
BLKMOV_B EN ENO IN OUT N	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	N	BYTE	要移动的字节数	V，I，Q，M，S，SM，L，AC，常数，＊VD，＊AC，＊LD
	OUT	BYTE	目的地首地址	V，I，Q，M，S，SM，L，AC，＊VD，＊LD，＊AC，常数（OUT 中无常数）
	IN	BYTE	源数据首地址	

【**例 4-34**】编写一段程序，将 VB0 开始的 4 个字节的内容传送至 VB10 开始的 4 个字节存储单元中，VB0～VB3 的数据分别为 5、6、7、8。

【**解**】程序运行结果如图 4-72 所示。

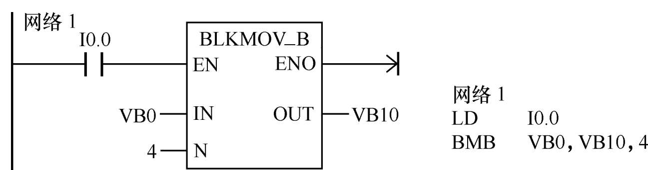


图 4-72　字节块传送程序示例

数组 1 的数据：5　6　7　8

数据地址：VB0　VB1　VB2　VB3

数组 2 的数据：5　6　7　8

数据地址：VB10　VB11　VB12　VB13

数据块传送指令还有块字传送和块双字传送，其使用方法和块字节传送类似，只不过其数据类型不同而已。

3. 字节交换指令（SWAP）

字节交换指令用来实现字中高、低字节内容的交换。当使能端（EN）输入有效时，将输入字 IN 中的高、低字节内容交换，结果仍放回字 IN 中。

其格式见表 4-23。

表 4-23　字节交换指令格式

LAD	参　　数	数据类型	说　　明	存　储　区
SWAP EN ENO IN	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN	WORD	源数据	V，I，Q，M，S，SM，T，C，L，AC，＊VD，＊AC，＊LD

【**例 4-35**】如图 4-73 所示的程序，若 QB0＝FF，QB1＝0，在接通 I0.0 的前后，PLC 的输出端的指示灯有何变化？

【**解**】

执行程序后，QB1＝FF，QB0＝0，因此运行程序前 PLC 的输出端的 QB0.0～QB0.7 指

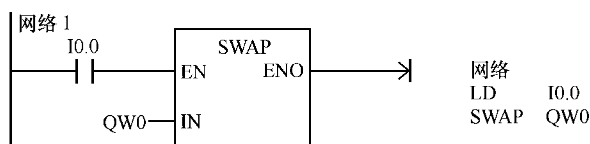


图 4-73　字节交换指令程序示例

示灯亮，执行程序后 QB0.0～QB0.7 指示灯灭，而 QB1.0～QB1.7 指示灯亮。

4. 字节填充指令（FILL）

字节填充指令用来实现存储器区域内容的填充。当使能端输入有效时，将输入字 IN 填充至从 OUT 指定单元开始的 *N* 个字存储单元。

字节填充指令可归类为表格处理指令，用于数据表的初始化，特别适合于连续字节的清 0，字节填充指令格式见表 4-24。

表 4-24　字节填充指令格式

LAD	参　数	数据类型	说　明	存　储　区
FILL_N EN ENO IN OUT N	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN	INT	要填充的数	V，I，Q，M，S，SM，L，T，C，AI，AC，常数，*VD，*LD，*AC
	OUT	INT	目的数据首地址	V，I，Q，M，S，SM，L，T，C，AQ，*VD，*LD，*AC
	N	BYTE	填充的个数	V，I，Q，M，S，SM，L，AC，常数，*VD，*LD，*AC

【例 4-36】 编写一段程序，将从 VW0 开始的 10 个字存储单元清 0。

【解】

程序如图 4-74 所示。FILL 是表指令，使用比较方便，特别是在程序的初始化时，常使用 FILL 指令将要用到的数据存储区清 0。在编写通信程序时，通常在程序的初始化时，将数据发送缓冲区和数据接收缓冲区的数据清 0，就要用到 FILL 指令。此外，表指令中还有 FIFO、LIFO 等指令，请读者参考相关手册。

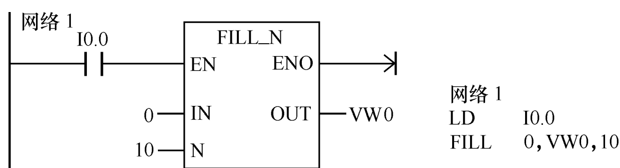


图 4-74　字节填充指令程序示例

当然也可以使用 BLKMOV 指令完成以上功能。

4.4.3　移位与循环指令

STEP 7 - Micro/WIN 提供的移位指令能将存储器的内容逐位向左或者向右移动。移动的位数由 *N* 决定。向左移 *N* 位相当于累加器的内容乘 2^N，向右移相当于累加器的内容除以 2^N。移位指令在逻辑控制中使用也很方便。移位与循环指令见表 4-25。

表 4-25　移位与循环指令汇总

名　称	语　句　表	梯　形　图	描　　述
字节左移	SLB	SHL_B	字节逐位左移，空出的位添 0
字左移	SLW	SHL_W	字逐位左移，空出的位添 0
双字左移	SLD	SHL_DW	双字逐位左移，空出的位添 0
字节右移	SRB	SHR_B	字节逐位右移，空出的位添 0
字右移	SRW	SHR_W	字逐位右移，空出的位添 0
双字右移	SRD	SHR_DW	双字逐位右移，空出的位添 0
字节循环左移	RLB	ROL_B	字节循环左移
字循环左移	RLW	ROL_W	字循环左移
双字循环左移	RLD	ROL_DW	双字循环左移
字节循环右移	RRB	ROR_B	字节循环右移
字循环右移	RRW	ROR_W	字循环右移
双字循环右移	RRD	ROR_DW	双字循环右移
移位寄存器	SHRB	SHRB	将 DATA 数值移入移位寄存器

1. 字左移（SHL_W）

当字左移指令（SHL_W）的 EN 位为高电平“1”时，执行移位指令，将 IN 端指定的内容左移 N 端指定的位数，然后写入 OUT 端指定的目的地地址中。如果移位数目（N）大于或等于 16，则数值最多被移位 16 次。最后一次移出的位保存在 SM1.1 中。字左移指令（SHL_W）和参数见表 4-26。

表 4-26　字左移指令（SHL_W）和参数

LAD	参　　数	数据类型	说　　明	存　储　区
SHL_W EN　ENO IN　OUT N	EN	BOOL	允许输入	I，Q，M，D，L
	ENO	BOOL	允许输出	
	N	BYTE	移动的位数	V，I，Q，M，S，SM，L，AC，常数，＊VD，＊LD，＊AC
	IN	WORD	移位对象	V，I，Q，M，S，SM，L，T，C，AC，＊VD，＊LD，＊AC，AI 和常数（OUT 无）
	OUT	WORD	移动操作结果	

【例 4-37】梯形图和指令表如图 4-75 所示。假设，IN 中的字 MW0 为 2#1001 1101 1111 1011，当 I0.0 闭合时，OUT 端的 MW0 中的数是多少？

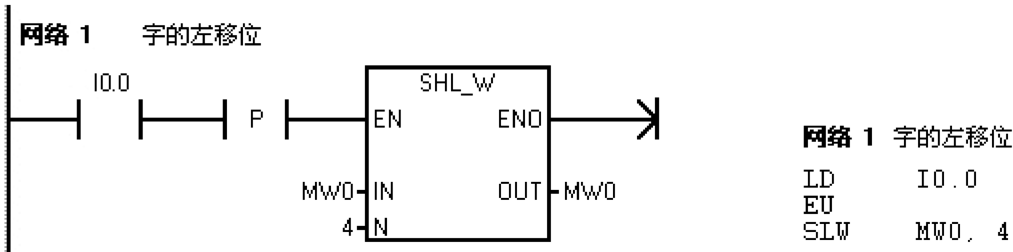


图 4-75　字左移指令应用举例

【解】

当 I0.0 闭合时，激活左移指令，IN 中的字存储在 MW0 中的数为 2#1001 1101 1111 1011，向左移 4 位后，OUT 端的 MW0 中的数是 2#1101 1111 1011 0000，字左移指令示意图如图 4-76 所示。

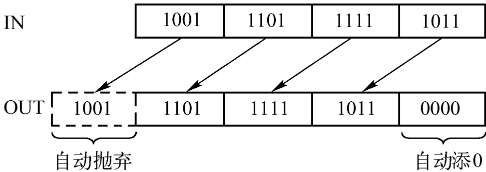


图 4-76　字左移指令示意图

【关键点】图 4-76 中的程序有一个上升沿，这样 I0.0 每闭合一次，左移 4 位，若没有上升沿，那么闭合一次，可能左移很多次。这点读者要特别注意。

2. 字右移（SHR_W）

当字右移指令（SHR_W）的 EN 位为高电平“1”时，将执行移位指令，将 IN 端指定的内容右移 N 端指定的位数，然后写入 OUT 端指定的目的地地址中。如果移位数目（N）大于或等于 16，则数值最多被移位 16 次。最后一次移出的位保存在 SM1.1 中。字右移指令（SHR_W）和参数见表 4-27。

表 4-27　字右移指令（SHR_W）和参数

LAD	参　数	数据类型	说　明	存　储　区
SHR_W EN　ENO IN　OUT N	EN	BOOL	允许输入	I，Q，M，S，L，V
	ENO	BOOL	允许输出	
	N	BYTE	移动的位数	V，I，Q，M，S，SM，L，AC，常数，＊VD，＊LD，＊AC
	IN	WORD	移位对象	V，I，Q，M，S，SM，L，T，C，AC，＊VD，＊LD，＊AC，AI 和常数（OUT 无）
	OUT	WORD	移动操作结果	

【**例 4-38**】梯形图和指令表如图 4-77 所示。假设 IN 中的字 MW0 为 2#1001 1101 1111 1011，当 I0.0 闭合时，OUT 端的 MW0 中的数是多少？

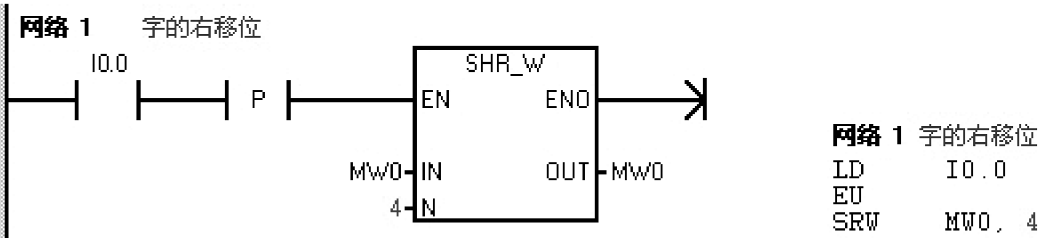


图 4-77　字右移指令应用举例

【解】

当 I0.0 闭合时，激活右移指令，IN 中的字存储在 MW0 中，假设这个数为 2#1001 1101

1111 1011，向右移 4 位后，OUT 端的 MW0 中的数是 2#0000 1001 1101 1111，字右移指令示意图如图 4-78 所示。

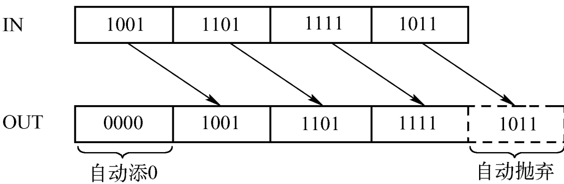


图 4-78　字右移指令示意图

【例 4-39】 设计一个 Q0.0 ~ Q0.7 的指示灯以 1 s 的周期依次点亮，然后熄灭，重复以上动作的程序。

【解】

程序如图 4-79 所示。

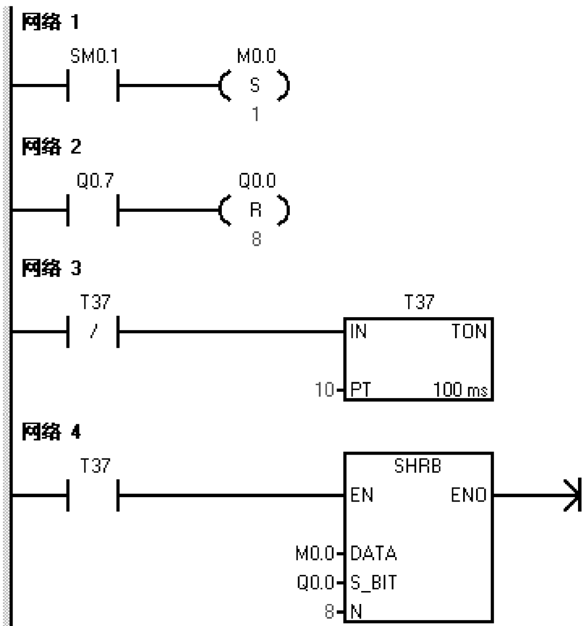


图 4-79　程序

字节的左移位、字节的右移位、双字的左移位、双字的右移位和字的移位指令类似，在此不再赘述。

3. 双字循环左移（ROL_DW）

当双字循环左移（ROL_DW）的 EN 位为高电平“1”时，将执行双字循环左移指令，将 IN 端指令的内容循环左移 N 端指定的位数，然后写入 OUT 端指令的目的地地址中。如果移位数目（N）大于或等于 32，执行循环之前在移动位数（N）上执行模数 32 操作。从而使移位数在 0 ~ 31 之间，例如当 N = 34 时，通过模运算，实际移位为 2。双字循环左移

（ROL_DW）指令和参数见表 4-28。

表 4-28　双字循环左移（ROL_DW）指令和参数

LAD	参　　数	数据类型	说　　明	存　储　区
ROL_DW EN ENO IN OUT N	EN	BOOL	允许输入	I，Q，M，S，L，V
	ENO	BOOL	允许输出	
	N	BYTE	移动的位数	V，I，Q，M，S，SM，L，AC，常数，＊VD，＊LD，＊AC
	IN	DWORD	移位对象	V，I，Q，M，S，SM，L，AC，＊VD，＊LD，＊AC，HC 和常数（OUT 无）
	OUT	DWORD	移动操作结果	

【例 4-40】 梯形图和指令表如图 4-80 所示。假设，IN 中的字 MD0 为 2#1001 1101 1111 1011 1001 1101 1111 1011，当 I0.0 闭合时，OUT 端的 MD0 中的数是多少?

【解】

当 I0.0 闭合时，激活双字循环左移指令，IN 中的双字存储在 MD0 中，除最高 4 位外，其余各位向左移 4 位后，双字的最高 4 位，循环到双字的最低 4 位，结果是 OUT 端的 MD0 中的数是 2#1101 1111 1011 1001 1101 1111 1011 1001，其示意图如图 4-81 所示。

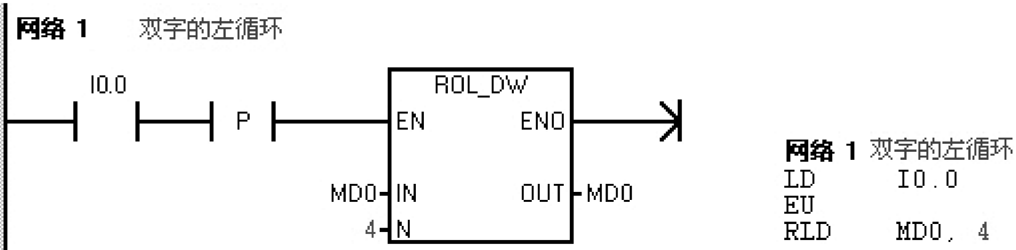


图 4-80　双字循环左移指令应用举例

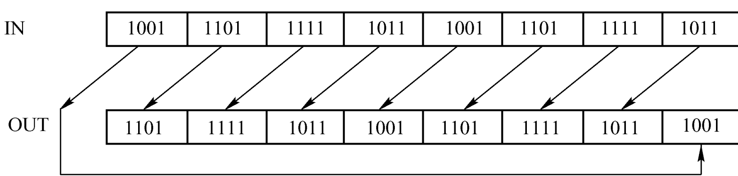


图 4-81　双字循环左移指令示意图

4. 双字循环右移（ROR_DW）

当双字循环右移（ROR_DW）的 EN 位为高电平“1”时，将执行双字循环右移指令，将 IN 端指令的内容向右循环移动 N 端指定的位数，然后写入 OUT 端指令的目的地地址中。如果移位数目（N）大于或等于 32，执行循环之前在移动位数（N）上执行模数 32 操作。从而使移位数在 0～31 之间，例如当 N＝34 时，通过模运算，实际移位为 2。双字循环右移（ROR_DW）指令和参数见表 4-29。

表 4-29　双字循环右移（ROR_DW）指令和参数

LAD	参　　数	数 据 类 型	说　　明	存　储　区
ROR_DW EN　ENO IN　OUT N	EN	BOOL	允许输入	I，Q，M，S，L，V
	ENO	BOOL	允许输出	
	N	BYTE	移动的位数	V，I，Q，M，S，SM，L，AC，常数，＊VD，＊LD，＊AC
	IN	DWORD	移位对象	V，I，Q，M，S，SM，L，AC，＊VD，＊LD，＊AC，HC 和常数（OUT 无）
	OUT	DWORD	移动操作结果	

【例 4-41】 梯形图和指令表如图 4-82 所示。假设 IN 中的双字 MD0 为 2#1001 1101 1111 1011 1001 1101 1111 1011，当 I0.0 闭合时，OUT 端的 MD0 中的数是多少?

【解】

当 I0.0 闭合时，激活双字循环右移指令，IN 中的双字存储在 MD0 中，这个数为 2#1001 1101 1111 1011 1001 1101 1111 1011，除最低 4 位外，其余各位向右移 4 位后，双字的最低 4 位，循环到双字的最高 4 位，结果是 OUT 端的 MD0 中的数是 2#1011 1001 1101 1111 1011 1001 1101 1111，其示意图如图 4-83 所示。

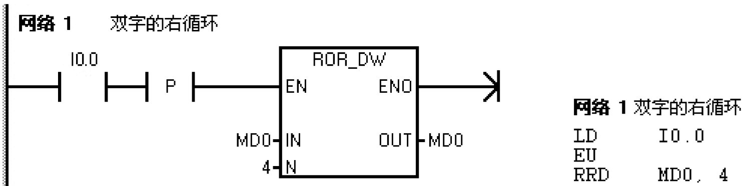


图 4-82　双字循环右移指令应用举例

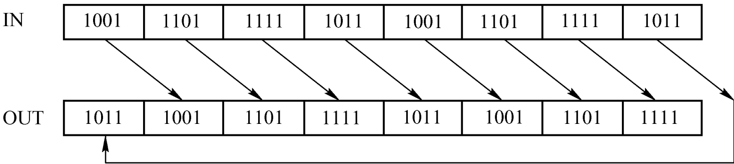


图 4-83　双字循环右移指令示意图

字节的左循环、字节的右循环、字的左循环、字的右循环和双字的循环指令类似，在此不再赘述。

4.4.4　时钟指令及其应用

1. 读取时钟指令及其应用

读取实时时钟指令（TODR）从硬件时钟中读当前时间和日期，并把它装载到一个 8 字节，起始地址为 T 的时间缓冲区中。设置实时时钟指令（TODW）将当前时间和日期写入硬件时钟，当前时钟存储在以地址 T 开始的 8 字节时间缓冲区中。必须按照 BCD 码的格式编码所有的日期和时间值（例如：用 16#97 表示 1997 年）。程序如图 4-84 所示。如果 PLC 系统的时间是 2009 年 4 月 8 日 8 时 6 分 5 秒，星期六，则运行的结果如图 4-85 所示。年份存入 VB0 存储单元，月份存入 VB1 单元，日存入 VB2 单元，小时存入 VB3 单元，分钟存入

VB4 单元，秒钟存入 VB5 单元，VB6 单元为 0，星期存入 VB7 单元，可见共占用 8 个存储单元。读取实时时钟（TODR）指令和参数见表 4-30。

表 4-30　读取实时时钟（TODR）指令和参数

LAD	参　数	数据类型	说　明	存　储　区
READ_RTC EN　ENO T	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	T	BYTE	存储日期的起始地址	V，I，Q，M，SM，S，L，＊VD，＊AC，＊LD

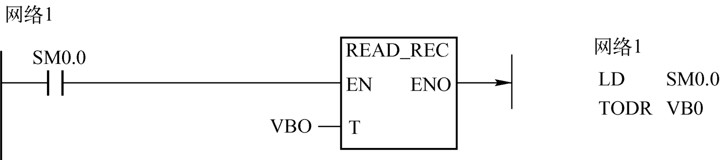


图 4-84　读取实时时钟指令应用举例

VB0	VB1	VB2	VB3	VB4	VB5	VB6	VB7
09	04	08	08	06	05	00	07

图 4-85　读取实时时钟指令的结果（BCD 码）

【关键点】 读取实时时钟（TODR）指令读取出来的日期是用 BCD 码表示，这点要特别注意。

2. 设置时钟指令及其应用

设置实时时钟（TODW）指令将当前时间和日期写入用 T 指定的在 8 个字节的时间缓冲区开始的硬件时钟。设置实时时钟（TODW）指令和参数见表 4-31。

表 4-31　设置实时时钟（TODW）指令和参数

LAD	参　数	数据类型	说　明	存　储　区
SET_RTC EN　ENO T	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	T	BYTE	存储日期的起始地址	V，I，Q，M，SM，S，L，＊VD，＊AC，＊LD

用一个例子说明设置时钟指令的应用。

【例 4-42】 把 2012 年 9 月 18 日 8 时 6 分 28 秒设置成 PLC 的当前时间。

【解】

先要做这样的设置：VB0 = 16#12，VB1 = 16#09，VB2 = 16#18，VB3 = 16#18，VB4 = 16#08，VB5 = 16#06，VB6 = 16#00，VB7 = 16#28，然后运行如图 4-86 所示的程序。

设置时钟还有一个简单的方法，不需要编写程序。只要进行简单设置即可，设置方法如下：

单击菜单栏中的“PLC”→“实时时钟”，如图 4-87 所示，弹出“时钟操作”界面，如图 4-88 所示，单击“读取 PC”按钮，读取计算机的当前时间，如图 4-89 所示。

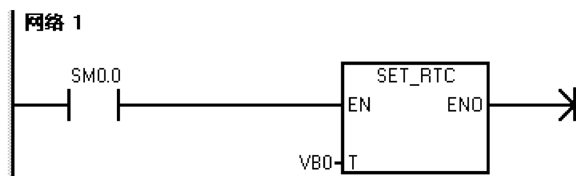


图 4-86　设置实时时钟指令实例



图 4-87　打开“时钟操作”界面

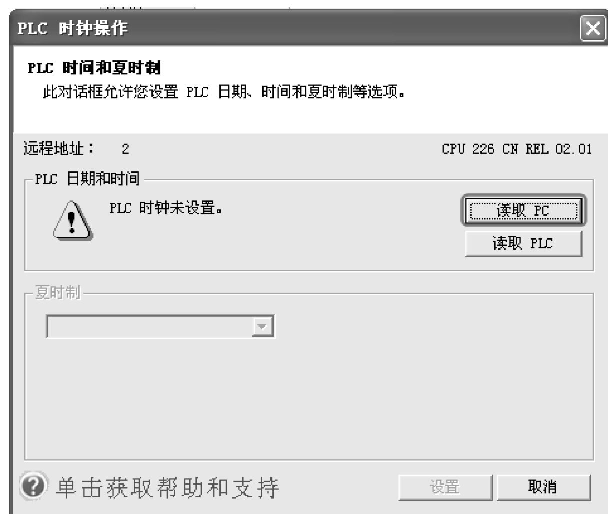


图 4-88　时钟操作界面

如图 4-89 所示，单击“设置”按钮可以将当前计算机的时间设置到 PLC 中，当然读者也可以设置其他时间。

【例 4-43】 某实验室的一个房间，要求每天 16∶30～18∶00 开启一个加热器，请用 PLC 实现此功能。

【解】

先用 PLC 读取实时时间，因为读取的时间是 BCD 码格式，所以之后要将 BCD 码转化成



图 4-89 设置实时时钟

整数，如果实时时间在 16:30～18:00，则开启加热器，梯形图如图 4-90 所示。

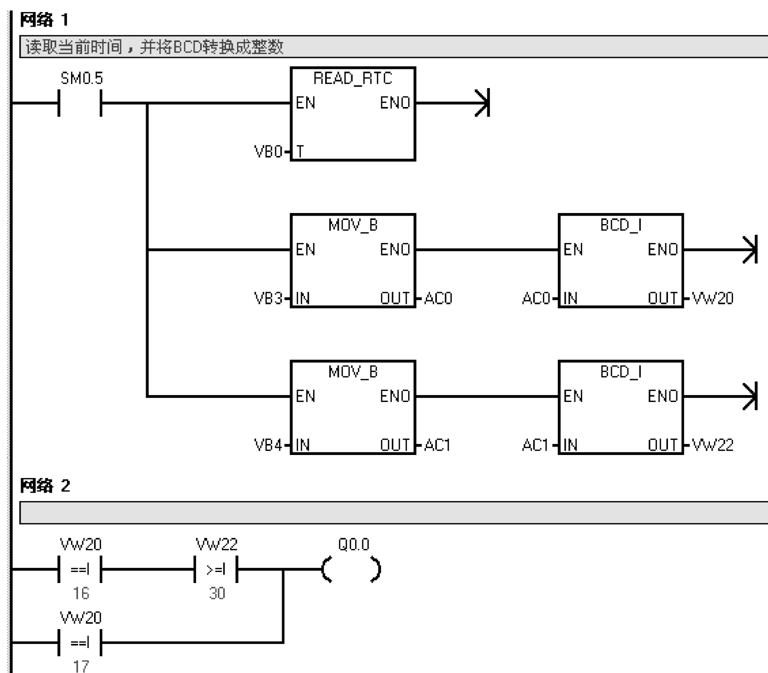


图 4-90 梯形图

4.4.5 算术运算指令

1. 整数算术运算指令

S7－200 的整数算术运算分为加法运算、减法运算、乘法运算和除法运算，其中每种运

算方式又有整数型和双整数型两种。

（1）整数加（ADD_I）

当允许输入端 EN 为高电平时，输入端 IN1 和 IN2 中的整数相加，结果送入 OUT 中。IN1 和 IN2 中的数可以是常数。整数加的表达式是：IN1 + IN2 = OUT。整数加（ADD_I）指令和参数见表 4-32。

表 4-32　整数加（ADD_I）指令和参数

LAD	参　数	数据类型	说　明	存　储　区
ADD_I EN　ENO IN1 IN2　OUT	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN1	INT	相加的第 1 个值	V，I，Q，M，S，SM，T，C，AC，L，AI，常数，*VD，*LD，*AC
	IN2	INT	相加的第 2 个值	
	OUT	INT	和	V，I，Q，M，S，SM，T，C，AC，L，*VD，*LD，*AC

【例 4-44】 梯形图和指令表如图 4-91 所示。MW0 中的整数为 11，MW2 中的整数为 21，则当 I0.0 闭合时，整数相加的结果存储在 OUT 端的 MW4 中，其结果是多少？

【解】

当 I0.0 闭合时，激活整数加指令，IN1 中的整数存储在 MW0 中，这个数为 11，IN2 中的整数存储在 MW2 中，这个数为 21，整数相加的结果存储在 OUT 端的 MW4 中的数是 32。由于没有超出计算范围，所以 Q0.0 输出为“1”。假设 IN1 中的整数为 9 999，IN2 中的整数为 30 000，则超过整数相加的范围。由于超出计算范围，所以 Q0.0 输出为“0”。

【关键点】 整数相加未超出范围时，当 I0.0 闭合，则 Q0.0 输出为高电平，否则 Q0.0 输出为低电平。

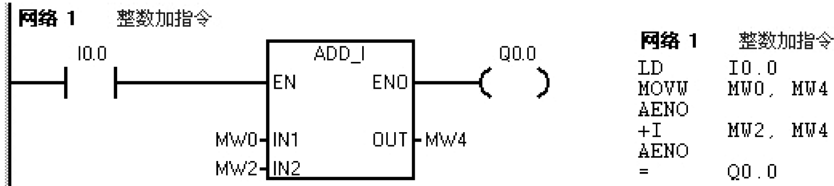


图 4-91　整数加（ADD_I）指令应用举例

双整数加（ADD_DI）指令与整数加（ADD_I）类似，只不过其数据类型为双整数，在此不再赘述。

（2）双整数减（SUB_DI）

当允许输入端 EN 为高电平时，输入端 IN1 和 IN2 中的双整数相减，结果送入 OUT 中。IN1 和 IN2 中的数可以是常数。双整数减的表达式是：IN1 - IN2 = OUT。双整数减（SUB_DI）指令和参数见表 4-33。

表 4-33　双整数减（SUB_DI）指令和参数

LAD	参　数	数据类型	说　明	存　储　区
SUB_DI EN　ENO IN1 IN2　OUT	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN1	DINT	被减数	V，I，Q，M，SM，S，L，AC，HC，常数，＊VD，＊LD，＊AC
	IN2	DINT	减数	
	OUT	DINT	差	V，I，Q，M，SM，S，L，AC，＊VD，＊LD，＊AC

【例 4-45】 梯形图和指令表如图 4-92 所示，IN1 中的双整数存储在 MD0 中，数值为 22，IN2 中的双整数存储在 MD4 中，数值为 11，当 I0.0 闭合时，双整数相减的结果存储在 OUT 端的 MD4 中，其结果是多少？

【解】

当 I0.0 闭合时，激活双整数减指令，IN1 中的双整数存储在 MD0 中，假设这个数为 22，IN2 中的双整数存储在 MD4 中，假设这个数为 11，双整数相减的结果存储在 OUT 端的 MD4 中的数是 11。由于没有超出计算范围，所以 Q0.0 输出为“1”。

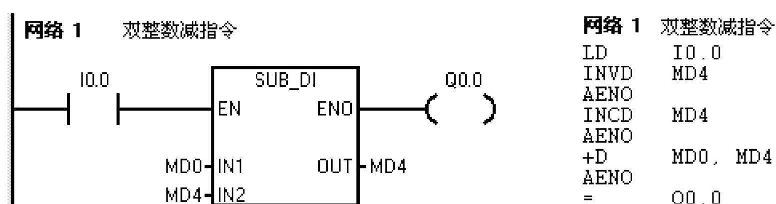


图 4-92　双整数减（SUB_DI）指令应用举例

整数减（SUB_I）指令与双整数减（SUB_DI）类似，只不过其数据类型为整数，在此不再赘述。

（3）整数乘（MUL_I）

当允许输入端 EN 为高电平时，输入端 IN1 和 IN2 中的整数相乘，结果送入 OUT 中。IN1 和 IN2 中的数可以是常数。整数乘的表达式是：IN1 × IN2 = OUT。整数乘（MUL_I）指令和参数见表 4-34。

表 4-34　整数乘（MUL_I）指令和参数

LAD	参　数	数据类型	说　明	存　储　区
MUL_I EN　ENO IN1 IN2　OUT	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN1	INT	相乘的第 1 个值	V，I，Q，M，S，SM，T，C，L，AC，AI，常数，＊VD，＊LD，＊AC
	IN2	INT	相乘的第 2 个值	
	OUT	INT	积	V，I，Q，M，S，SM，L，T，C，AC，＊VD，＊LD，＊AC

【例 4-46】 梯形图和指令表如图 4-93 所示。IN1 中的整数存储在 MW0 中，数值为 11，IN2 中的整数存储在 MW2 中，数值为 11，当 I0.0 闭合时，整数相乘的结果存储在 OUT 端

的 MW4 中，其结果是多少？

【解】

当 I0.0 闭合时，激活整数乘指令，IN1 × IN2 = OUT，整数相乘的结果存储在 OUT 端的 MW4 中，结果是 121。由于没有超出计算范围，所以 Q0.0 输出为“1”。

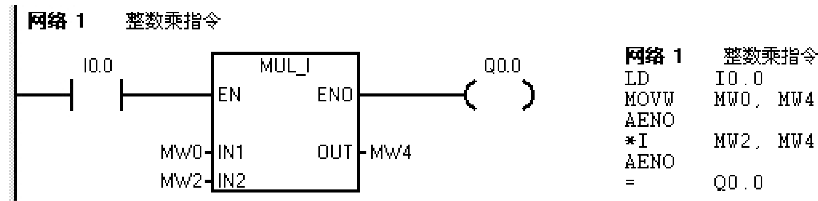


图 4-93　整数乘（MUL_I）指令应用举例

两个整数相乘得双整数乘指令（MUL），其两个乘数都是整数，乘积为双整数，注意 MUL 和 MUL_I 的区别。

双整数乘（MUL_DI）指令与整数乘（MUL_I）类似，只不过双整数乘数据类型为双整数，在此不再赘述。

（4）双整数除（DIV_DI）

当允许输入端 EN 为高电平时，输入端 IN1 中的双整数除以 IN2 中的双整数，结果为双整数，送入 OUT 中，不保留余数。IN1 和 IN2 中的数可以是常数。双整数除（DIV_DI）指令和参数见表 4-35。

表 4-35　双整数除（DIV_DI）指令和参数

LAD	参　数	数据类型	说　明	存　储　区
DIV_DI: EN ENO, IN1, IN2 OUT	EN	BOOL	允许输入	V, I, Q, M, S, SM, L
	ENO	BOOL	允许输出	
	IN1	DINT	被除数	V, I, Q, M, SM, S, L, HC, AC, 常数, * VD, * LD, * AC
	IN2	DINT	除数	
	OUT	DINT	商	V, I, Q, M, SM, S, L, AC, * VD, * LD, * AC

【**例 4-47**】梯形图和指令表如图 4-94 所示。IN1 中的双整数存储在 MD0 中，数值为 11，IN2 中的双整数存储在 MD4 中，数值为 2，当 I0.0 闭合时，双整数相除的结果存储在 OUT 端的 MD8 中，其结果是多少？

【解】

当 I0.0 闭合时，激活双整数除指令，IN1 中的双整数存储在 MD0 中，数值为 11，IN2 中的双整数存储在 MD4 中，数值为 2，双整数相除的结果存储在 OUT 端的 MD8 中的数是 5，不产生余数。由于没有超出计算范围，所以 Q0.0 输出为“1”。

【**关键点**】双整数除法不产生余数。

整数除（DIV_I）指令与双整数除（DIV_DI）类似，只不过其数据类型为整数，在此不再赘述。整数相除得商和余数指令（DIV），其除数和被除数都是整数，输出 OUT 为双整数，其高位是一个 16 位余数，其低位是一个 16 位商，注意 DIV 和 DIV_I 的区别。

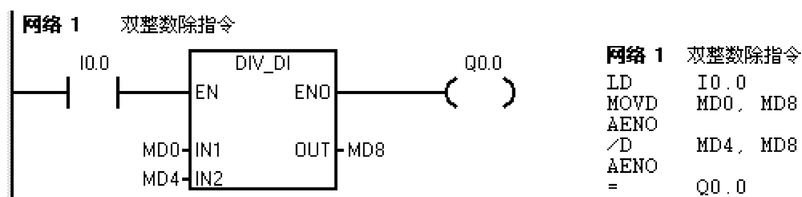


图 4-94　双整数除（DIV_DI）指令应用举例

【例 4-48】算术运算程序示例，其中开始时 AC1 中内容为 4 000，AC0 中内容为 6 000，VD100 中内容为 200，VW200 中内容为 41，程序和运行结果如图 4-95 所示。

【解】

累加器 AC0 和 AC1 中可以装入字节、字、双字和实数等数据类型的数据，可见其使用比较灵活。DIV 指令的除数和被除数都是整数，而结果为双整数，对于本例除数为 4 000，被除数 41，双整数结果存储在 VD202 中，其中余数 23 存储在高位 VW202 中，商 97 存储在低位 VW204 中。

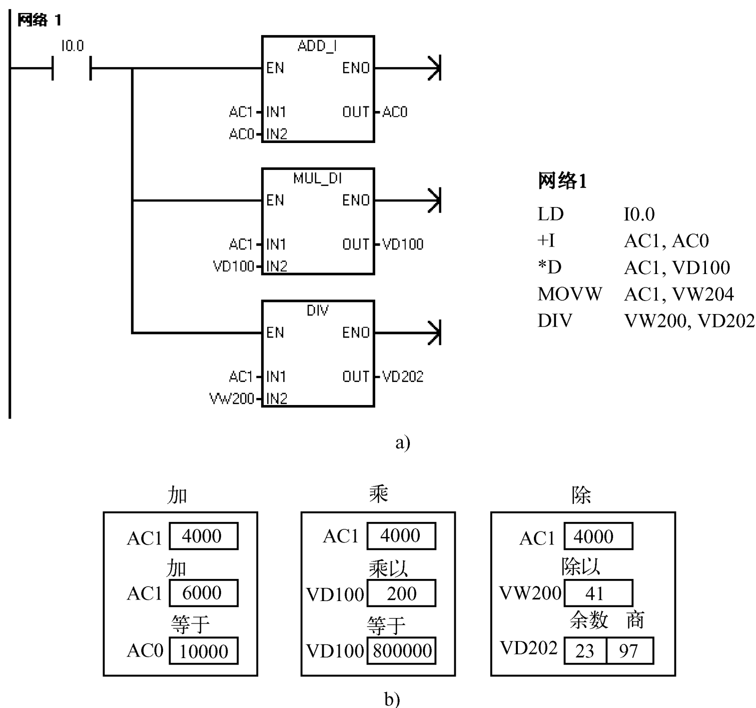


图 4-95　程序和运行结果
a）程序　b）运行结果

【例 4-49】用模拟电位器调节定时器 T37 的设定值为 5～20 s，设计此程序。

【解】

CPU 221 和 CPU 222 有一个模拟电位器，其他 CPU 有 2 个模拟电位器。CPU 将电位器的位置转换为 0～255 的数值，然后存入 SM28 和 SM29 中，分别对应电位器 0 和电位器 1 的值。电位器的位置用小螺钉旋具调整。

由于设定时间的范围是 5 ~ 20 s，电位器上对应的数字是 0 ~ 255，设读出的数字为 X，则 100 ms 定时器（单位是 0.1 ms）的设定值为

$$(200-50)\times X/255+50=150\times X/255+50$$

为了保证精度，要先乘法后除法，梯形图如图 4-96 所示。

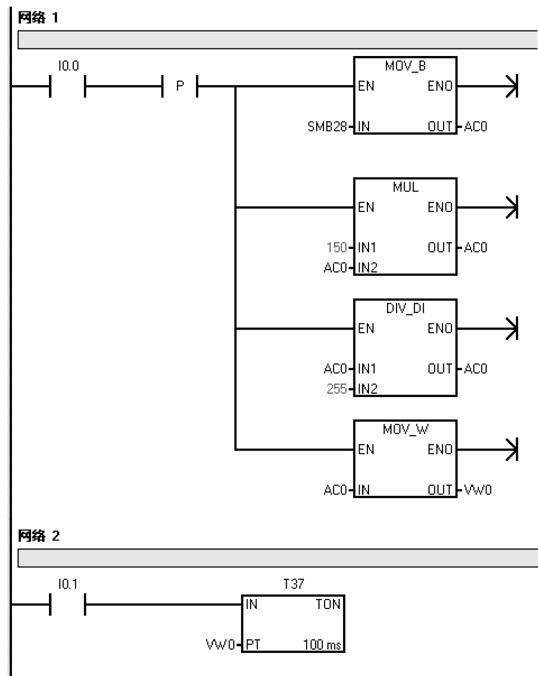


图 4-96　梯形图

（5）递增/递减运算指令

在输入端 IN1 上加 1 或减 1，并将结果置入 OUT。递增/递减指令的操作数类型为字节、字和双字。字递增运算指令格式见表 4-36。

表 4-36　字递增运算指令格式

LAD	参　　数	数 据 类 型	说　　明	存　储　区
INC_W EN　ENO IN　OUT	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN1	INT	将要递增 1 的数	V，I，Q，M，S，SM，AC，AI，L，T，C，常数，* VD，* LD，* AC
	OUT	INT	递增 1 后的结果	V，I，Q，M，S，SM，L，AC，T，C，* VD，* LD，* AC

1）字节递增/字节递减运算（INC_B/DEC_B）。使能端输入有效时，将一个字节的无符号数 IN 增 1/减 1，并将结果送至 OUT 指定的存储器单元输出。

2）双字递增/双字递减运算（INC_DW/DEC_DW）。使能端输入有效时，将双字长的符号数 IN 增 1/减 1，并将结果送至 OUT 指定的存储器单元输出。

【例 4-50】 递增/递减运算程序如图 4-97a 所示。初始时 AC0 中的内容为 125，VD100 中的内容为 128 000，试分析运算结果。

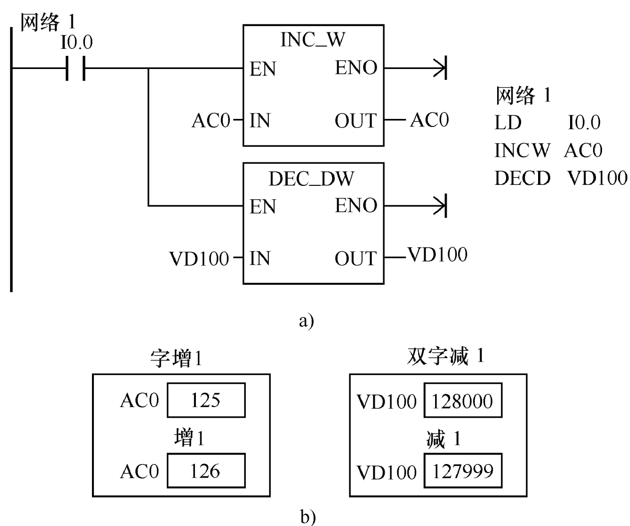


图 4-97　程序和运行结果
a）程序　b）运行结果

【例 4-51】 设计记录一台设备的运行的时间，当设备运行时，I0.0 为 1，停止时 I0.0 为 0，测量的小时数存放在 VW0 中，分钟数存放在 VW2，秒数存放在 VW4 中，当前的秒数显示在数码管上。

【解】

程序如图 4-98 所示。

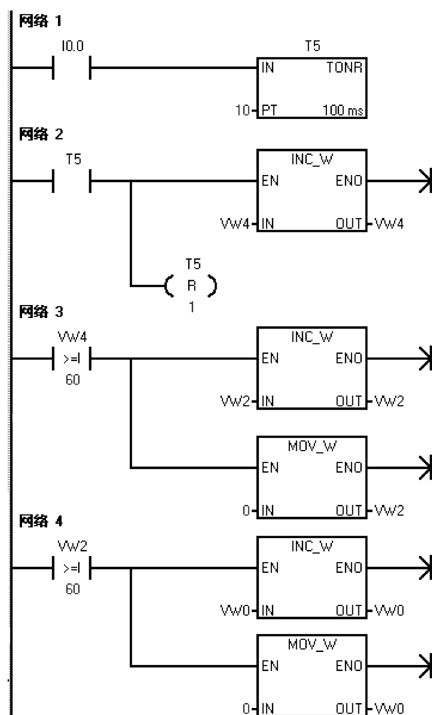


图 4-98　程序

【例 4-52】有一个电炉，加热功率有 1 000 W、2 000 W 和 3 000 W 三个档次，电炉有 1 000 W 和 2 000 W 两种电加热丝。要求用一个按钮选择三个加热档，当按一次按钮时，1 000 W 电阻丝加热，即第一档；当按两次按钮时，2 000 W 电阻丝加热，即第二档；当按三次按钮时，1 000 W 和 2 000 W 电阻丝同时加热，即第三档；当按四次按钮时停止加热，请编写程序。

【解】

程序如图 4-99 所示。

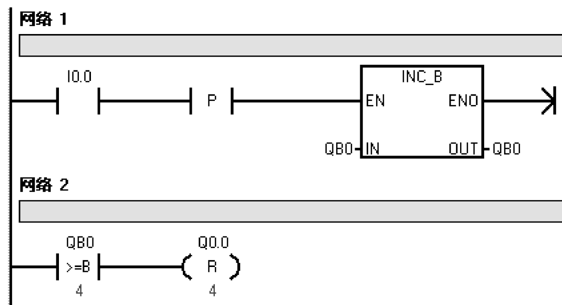


图 4-99　程序

2. 浮点数运算指令

浮点数函数有浮点数运算函数、三角函数、对数函数、幂运算函数和 PID 等。浮点数运算函数又分为加法运算、减法运算、乘法运算和除法运算函数。浮点数运算函数见表 4-37。

表 4-37　浮点数运算函数

语句表	梯形图	描述
+R	ADD_R	将两个 32 位实数相加，并产生一个 32 位实数结果（OUT）
-R	SUB_R	将两个 32 位实数相减，并产生一个 32 位实数结果（OUT）
*R	MUL_R	将两个 32 位实数相乘，并产生一个 32 位实数结果（OUT）
/R	DIV_R	将两个 32 位实数相除，并产生一个 32 位实数商
SQRT	SQRT	求浮点数的平方根
EXP	EXP	求浮点数的自然指数
LN	LN	求浮点数的自然对数
SIN	SIN	求浮点数的正弦函数
COS	COS	求浮点数的余弦函数
TAN	TAN	求浮点数的正切函数
PID	PID	PID 运算

分别介绍如下：

① 实数加（ADD_R）。当允许输入端 EN 为高电平时，输入端 IN1 和 IN2 中的实数相加，结果送入 OUT 中。IN1 和 IN2 中的数可以是常数。实数加的表达式是：IN1 + IN2 = OUT。实数加（ADD_R）指令和参数见表 4-38。

表 4-38　实数加（ADD_R）指令和参数

LAD	参　数	数据类型	说　明	存　储　区
ADD_R EN ENO IN1 IN2 OUT	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN1	REAL	相加的第 1 个值	V，I，Q，M，S，SM，L，AC，常数，*VD，*LD，*AC
	IN2	REAL	相加的第 2 个值	
	OUT	REAL	相加的结果（和）	V，I，Q，M，S，SM，L，AC，*VD，*LD，*AC

用一个例子来说明实数加（ADD_R）指令，梯形图和指令表如图 4-100 所示。当 I0.0 闭合时，激活实数加指令，IN1 中的实数存储在 MD0 中，假设这个数为 10.1，IN2 中的实数存储在 MD4 中，假设这个数为 21.1，实数相加的结果存储在 OUT 端的 MD8 中的数是 31.2。

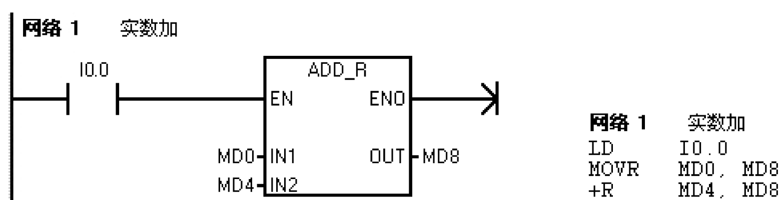


图 4-100　实数加（ADD_R）指令应用举例

② 实数减（SUB_R）、实数乘（MUL_R）和实数除（DIV_R）的使用方法与前面的指令用法类似，在此不再赘述。

③ MUL_DI/DIV_DI 和 MUL_R/DIV_R 的输入都是 32 位，输出的结果也是 32 位，但前者的输入和输出是双整数，属于双整数运算，而后者输入和输出的是实数，属于浮点运算，简单地说，后者的输入和输出数据中有小数点，而前者没有，后者的运算速度要慢得多。

值得注意的是，乘/除运算对特殊标志位 SM1.0（零标志位）、SM1.1（溢出标志位）、SM1.2（负数标志位）、SM1.3（被 0 除标志位）会产生影响。若 SM1.1 在乘法运算中被置 1，表明结果溢出，则其他标志位状态均置 0，无输出。若 SM1.3 在除法运算中被置 1，说明除数为 0，则其他标志位状态保持不变，原操作数也不变。

【关键点】 浮点数的算术指令的输入端可以是常数，但必须是带有小数点的常数，如 5.0，不能为 5，否则会出错。

3. 转换指令

转换指令是将一种数据格式转换成另外一种格式进行存储。例如，要让一个整型数据和双整型数据进行算术运算，一般要将整型数据转换成双整型数据。STEP 7 - Micro/Win 的转换指令见表 4-39。

表 4-39　转换指令

STL	LAD	说　明
BTI	B_I	将字节数值（IN）转换成整数值，并将结果置入 OUT 指定的变量中
ITB	I_B	将整数值（IN）转换成字节值，并将结果置入 OUT 指定的变量中

（续）

STL	LAD	说　　明
ITD	I_DI	将整数值（IN）转换成双整数值，并将结果置入 OUT 指定的变量中
ITS	I_S	将整数值（IN）转换为长度为 8 个字符的 ASCII 字符串
DTI	DI_I	双整数值（IN）转换成整数值，并将结果置入 OUT 指定的变量中
DTR	DI_R	将 32 位带符号整数（IN）转换成 32 位实数，并将结果置入 OUT 指定的变量中
DTS	DI_S	将双整数（IN）转换为长度为 12 个字符的 ASCII 字符串
BTI	BCD_I	将二进制编码的十进制值（IN）转换成整数值，并将结果载入 OUT 指定的变量中
ITB	I_BCD	将输入整数值（IN）转换成二进制编码的十进制数，并将结果载入 OUT 指定的变量中
RND	ROUND	将实值（IN）转换成双整数值，并将结果置入 OUT 指定的变量中
TRUNC	TRUNC	将 32 位实数（IN）转换成 32 位双整数，并将结果的整数部分置入 OUT 指定的变量中
RTS	R_S	将实数值（IN）转换为 ASCII 字符串
ITA	ITA	将整数值（IN）转换成 ASCII 字符数组
DTA	DTA	将双字（IN）转换成 ASCII 字符数组
RTA	RTA	将实数值（IN）转换成 ASCII 字符
ATH	ATH	将从 IN 开始的 ASCII 字符号码（LEN）转换成从 OUT 开始的十六进制数字
HTA	HTA	将从 IN 开始的 ASCII 字符号码（LEN）转换成从 OUT 开始的十六进制数字
STI	S_I	将字符串数值（IN）转换为存储在 OUT 中的整数值，从偏移量 INDX 位置开始
STD	S_DI	将字符串值（IN）转换为存储在 OUT 中的双整数值，从偏移量 INDX 位置开始
STR	S_R	将字符串值（IN）转换为存储在 OUT 中的实数值，从偏移量 INDX 位置开始
DECO	DECO	设置输出字（OUT）中与用输入字节（IN）最低“半字节”（4 位）表示的位数相对应的位
ENCO	ENCO	将输入字（IN）最低位的位数写入输出字节（OUT）的最低“半字节”（4 个位）中
SEG	SEG	生成照明七段显示段的位格式

（1）整数转换成双整数（ITD）

整数转换成双整数指令是将 IN 端指定的内容以整数的格式读入，然后将其转换为双整数码格式输出到 OUT 端。整数转换成双整数指令和参数见表 4-40。

表 4-40　整数转换成双整数指令和参数

LAD	参　　数	数据类型	说　　明	存　储　区
I_DI EN　ENO IN　OUT	EN	BOOL	使能（允许输入）	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN	INT	输入的整数	V，I，Q，M，S，SM，L，T，C，AI，AC，常数，＊VD，＊LD，＊AC
	OUT	DINT	整数转化成的双整数	V，I，Q，M，S，SM，L，AC，＊VD，＊LD，＊AC

【例 4-53】 梯形图和指令表如图 4-101 所示。IN 中的整数存储在 MW0 中（用十六进制表示为 16#0016），当 I0.0 闭合时，转换完成后 OUT 端的 MD2 中的双整数是多少？

【解】

当 I0.0 闭合时，激活整数转换成双整数指令，IN 中的整数存储在 MW0 中（用十六进制表示为 16#0016），转换完成后 OUT 端的 MD2 中的双整数是 16#0000 0016。但要注意，MW2 = 16#0000，而 MW4 = 16#0016。

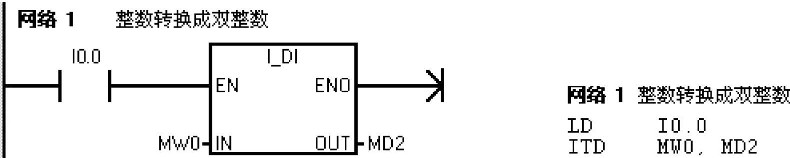


图 4-101 整数转换成双整数指令应用举例

（2）双整数转换成实数（DTR）

双整数转换成实数指令是将 IN 端指定的内容以双整数的格式读入，然后将其转换为实数码格式输出到 OUT 端。实数格式在后续算术计算中是很常用的，如 3.14 就是实数形式。双整数转换成实数指令和参数见表 4-41。

表 4-41 双整数转换成实数指令和参数

LAD	参 数	数据类型	说 明	存 储 区
DI_R EN ENO IN OUT	EN	BOOL	使能（允许输入）	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN	DINT	输入的双整数	V，I，Q，M，S，SM，L，HC，AC，常数，* VD，* AC，* LD
	OUT	REAL	双整数转化成的实数	V，I，Q，M，S，SM，L，AC，* VD，* LD，* AC

【例 4-54】 梯形图和指令表如图 4-102 所示。IN 中的双整数存储在 MD0 中，（用十进制表示为 16），转换完成后 OUT 端的 MD4 中的实数是多少？

【解】

当 I0.0 闭合时，激活双整数转换成实数指令，IN 中的双整数存储在 MD0 中（用十进制表示为 16），转换完成后 OUT 端的 MD4 中的实数是 16.0。一个实数要用 4 个字节存储。

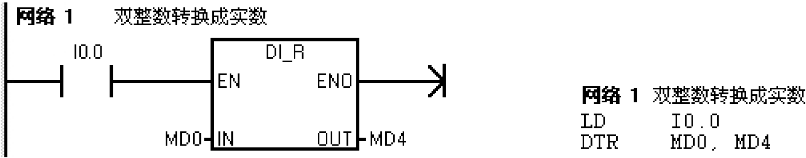


图 4-102 双整数转换成实数指令应用举例

【关键点】 应用 I_DI 转换指令后，数值的大小并未改变，但有时转换是必需的，因为只有相同的数据类型，才可以进行数学运算，例如要将一个整数和双整数相加，则比较保险的做法是先将整数转化成双整数，再做双整数加法。

DI_I 是双整数转换成整数的指令，并将结果存入 OUT 指定的变量中。若双整数太大，则会溢出。

DI_R 是双整数转换成实数的指令，并将结果存入 OUT 指定的变量中。

（3）实数四舍五入为双整数（ROUND）

ROUND 指令是将实数进行四舍五入取整后转换成双整数的格式。实数四舍五入为双整数指令和参数见表 4-42。

表 4-42　实数四舍五入为双整数指令和参数

LAD	参　　数	数据类型	说　　明	存　储　区
ROUND EN ENO IN OUT	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN	REAL	实数（浮点型）	V，I，Q，M，S，SM，L，AC，常数，＊VD，＊LD，＊AC
	OUT	DINT	四舍五入后为双整数	V，I，Q，M，S，SM，L，AC，＊VD，＊LD，＊AC

【例 4-55】 梯形图和指令表如图 4-103 所示。IN 中的实数存储在 MD0 中，假设这个实数为 3. 14，进行四舍五入运算后 OUT 端的 MD4 中的双整数是多少？假设这个实数为 3. 88，进行四舍五入运算后 OUT 端的 MD4 中的双整数是多少？

【解】

当 I0. 0 闭合时，激活实数四舍五入指令，IN 中的实数存储在 MD0 中，假设这个实数为 3. 14，进行四舍五入运算后 OUT 端的 MD4 中的双整数是 3，假设这个实数为 3. 88，进行四舍五入运算后 OUT 端的 MD4 中的双整数是 4。

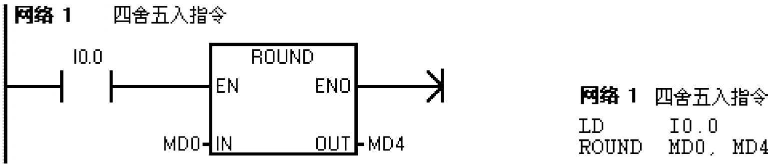


图 4-103　实数四舍五入为双整数指令应用举例

【关键点】 ROUND 是四舍五入指令，而 TRUNC 是取整指令，将输入的 32 位实数转换成整数，只有整数部分保留，舍去小数部分，结果为双整数，并将结果存入 OUT 指定的变量中。例如输入是 32. 2，执行 ROUND 或者 TRUNC 指令，结果转换成 32。而输入是 32. 5，执行 TRUNC 指令，结果转换成 32；执行 ROUND 指令，结果转换成 33。请注意区分。

【例 4-56】 将英寸转换成厘米，已知单位为英寸的长度保存在 VW0 中，数据类型为整数，英寸和厘米的转换单位为 2. 54，数据类型为实数，要将最终单位厘米的结果保存在 VD4 中，且结果为整数。编写程序实现这一功能。

【解】

要将单位为英寸的长度转化成单位为厘米的长度，必须要用到实数乘法，因此乘数必须为实数，而已知的英寸长度是整数，所以先要将整数转换成双整数，再将双整数转换成实数，最后将乘积取整就得到结果。程序如图 4-104 所示。

4. 数学功能指令

数学功能指令包含正弦（SIN）、余弦（COS）、正切（TAN）、自然对数（LN）、自然

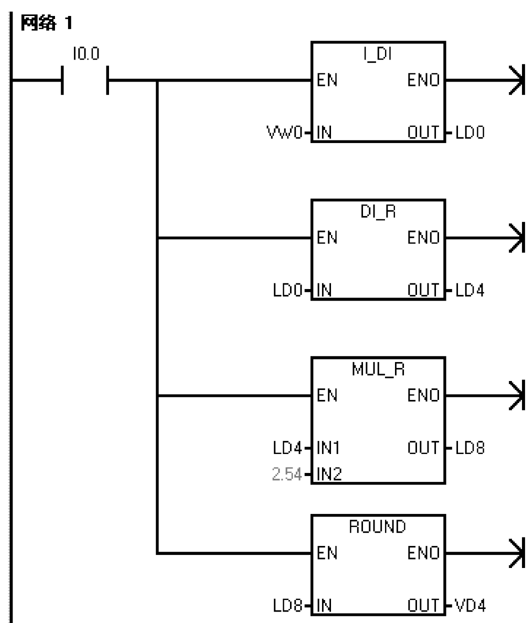


图 4-104　程序

指数（EXP）和平方根（SQRT）等。这些指令的使用比较简单，仅以正弦（SIN）为例说明数学功能指令的使用，见表 4-43。

表 4-43　求正弦（SIN）值指令和参数

LAD	参　　数	数据类型	说　　明	存　储　区
SIN EN ENO IN OUT	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN	REAL	输入值	V，I，Q，M，SM，S，L，AC，常数，* VD，* LD，* AC
	OUT	REAL	输出值（正弦值）	V，I，Q，M，SM，S，L，AC，* VD，* LD，* AC

用一个例子来说明求正弦（SIN）值指令，梯形图和指令表如图 4-105 所示。当 I0.0 闭合时，激活求正弦值指令，IN 中的实数存储在 VD0 中，假设这个数为 0.5，实数求正弦的结果存储在 OUT 端的 VD8 中的数是 0.479。

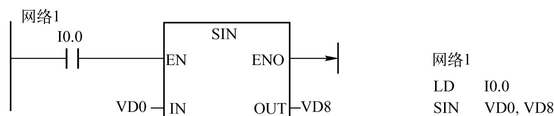


图 4-105　正弦运算指令应用示例

【关键点】 三角函数的输入值是弧度，而不是角度。

求余弦（COS）和求正切（TAN）的使用方法与前面的指令用法类似，在此不再赘述。

5. 编码和解码指令

编码指令（ENCO）将输入字 IN 的最低有效位的位号写入输出字 OUT 的最低有效“半字节”（4 位）中。解码指令（DECO）根据输入字 IN 的低 4 位所表示的位号，置输出字 OUT 的相应位为 1。也有人称解码指令为译码指令。编码和解码指令格式见表 4-44。

表 4-44　编码和解码指令格式

LAD	参　数	数据类型	说　明	存储区
ENCO（EN ENO IN OUT）	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN	WORD	输入值	V，I，Q，M，SM，L，S，AQ，T，C，AC，＊VD，＊AC，＊LD
	OUT	BYTE	输出值	V，I，Q，M，SM，S，L，AC，常数，＊VD，＊LD，＊AC
DECO（EN ENO IN OUT）	EN	BOOL	允许输入	V，I，Q，M，S，SM，L
	ENO	BOOL	允许输出	
	IN	BYTE	输入值	V，I，Q，M，SM，S，L，AC，常数，＊VD，＊LD，＊AC
	OUT	WORD	输出值	V，I，Q，M，SM，L，S，AQ，T，C，AC，＊VD，＊AC，＊LD

用一个例子说明以上指令的应用，如图 4-106 所示是编码和解码指令程序示例。

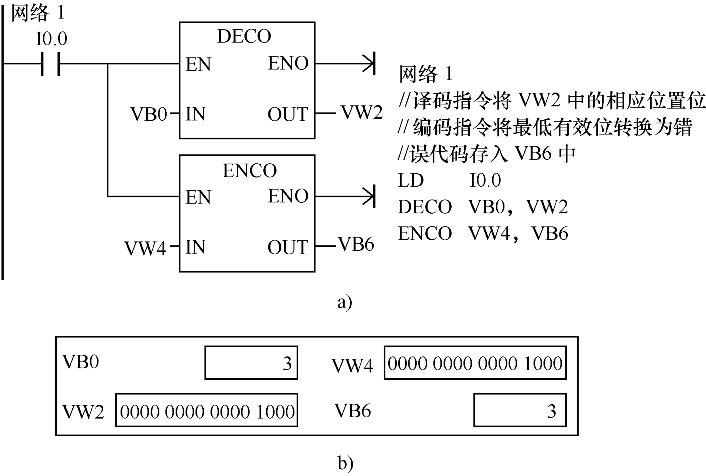


图 4-106　编码和解码指令程序示例
a）程序　b）运行结果

4.4.6　功能指令的应用

功能指令主要用于数字运算及处理场合，完成运算、数据的生成、存储以及某些规律的任务实现。功能指令除了能处理以上特殊功能外，也可用于逻辑控制程序中，这为逻辑控制类编程提供了新思路。

【例 4-57】用功能指令编写程序，有 5 台电动机，接在 Q0.1 ~ Q0.5 的输出接线端子上，使用单按钮控制起/停。按钮接在 I0.0 上，具体的控制方法是，按下按钮的次数对应起动电动机的号码，最后按下按钮持续 3 s，电动机停止。

【解】

梯形图如图 4-107 所示。

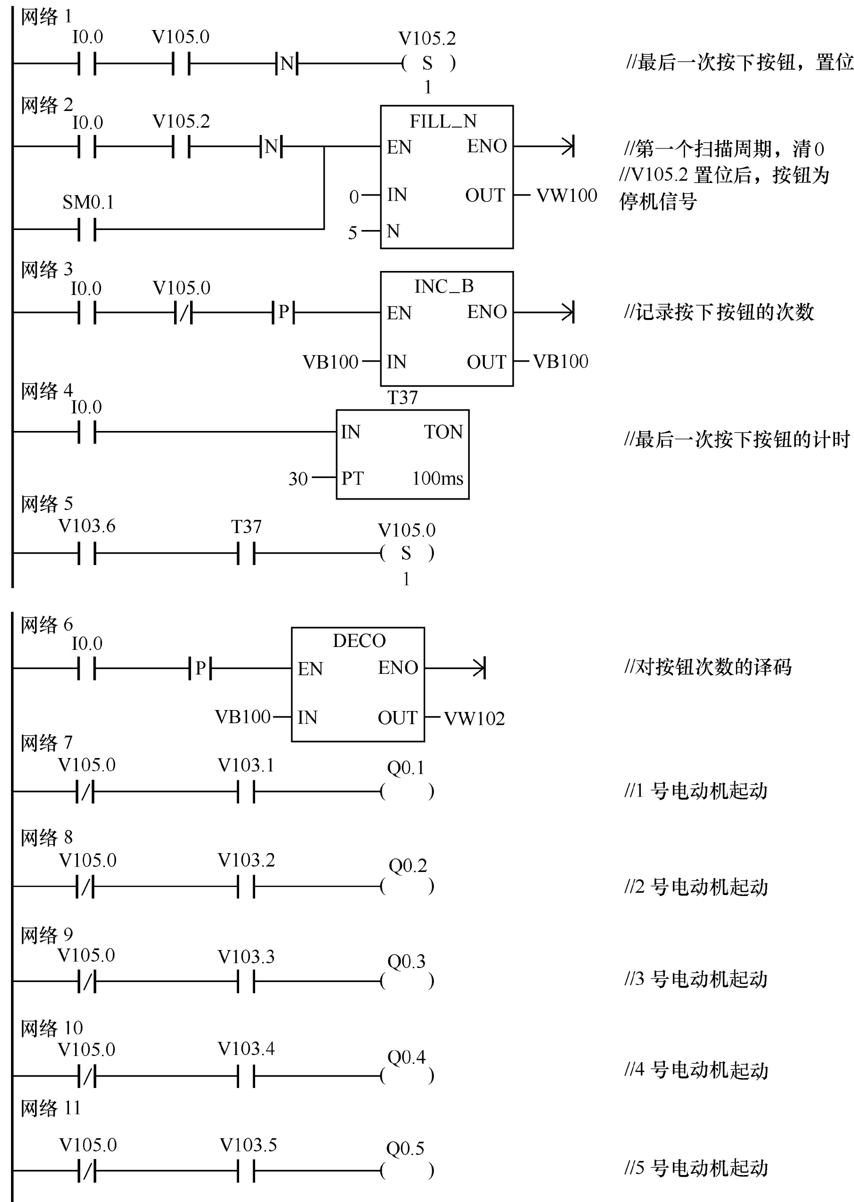
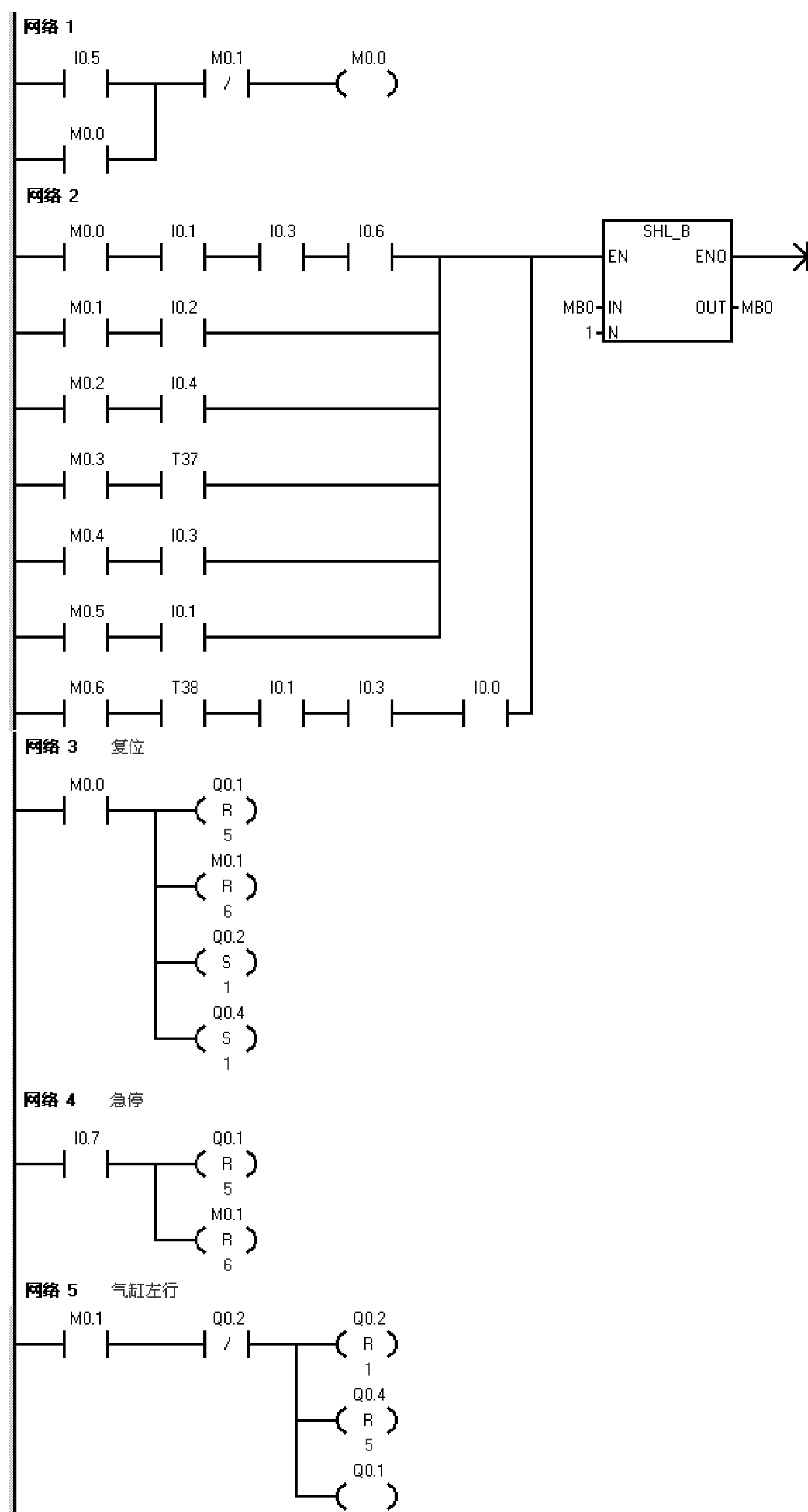


图 4-107　程序

【例 4-58】用功能指令编写程序，控制要求与例 4-30 相同。

【解】

梯形图如图 4-108 所示。


网络 1
I0.5
M0.1
M0.0
M0.0
网络 2
M0.0
I0.1
I0.3
I0.6
SHL_B
EN
ENO
MB0
IN
OUT
MB0
1
N
M0.1
I0.2
M0.2
I0.4
M0.3
T37
M0.4
I0.3
M0.5
I0.1
M0.6
T38
I0.1
I0.3
I0.0
网络 3 复位
M0.0
Q0.1
R
5
M0.1
R
6
Q0.2
S
1
Q0.4
S
1
网络 4 急停
I0.7
Q0.1
R
5
M0.1
R
6
网络 5 气缸左行
M0.1
Q0.2
Q0.2
R
1
Q0.4
R
5
Q0.1
图 4-108　机械手程序

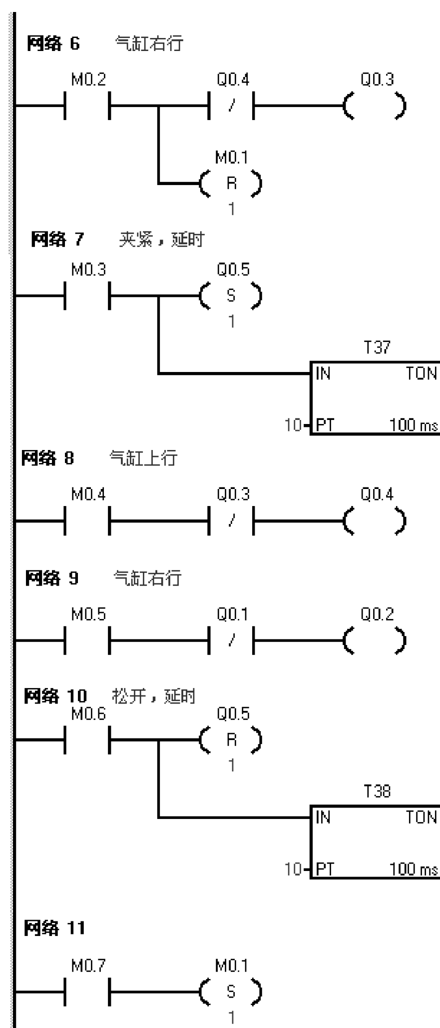


图 4-108　机械手程序（续）

【关键点】理解移位指令是解题的关键。

4.5　S7－200 PLC 的程序控制指令及其应用

程序控制指令包含跳转指令、循环指令、子程序指令、中断指令和顺控继电器指令。程序控制指令用于程序执行流程的控制。对于一个扫描周期而言，跳转指令可以使程序出现跳跃以实现网络的选择；子程序指令可调用某些子程序，增强程序的结构化，使程序的可读性增强，使程序更加简洁；中断指令则是用于中断信号引起的子程序调用；顺控继电器指令可形成状态网络中各状态的激活及隔离。

4.5.1　子程序调用

通常将具有特定功能，并且能多次使用的程序段作为子程序。子程序可以多次被调用，也可以嵌套（最多 8 层）。

子程序有子程序调用和子程序返回两大类指令，子程序返回又分条件返回和无条件返回。子程序调用指令用在主程序或其他调用子程序的程序中，子程序的无条件返回指令用在子程序的最后网络段。子程序结束时，程序执行应返回原调用指令（CALL）的下一条指令处。

建立子程序的方法：在编程软件的程序编辑窗口的下方有主程序（OB1）、子程序（SBR0）、中断服务程序（INT0）的标签，单击“子程序”选项即可进入SBR0子程序显示区，也可以通过指令树的项目进入子程序SBR0显示区。添加一个子程序时，可以用编辑菜单的插入项增加一个子程序，子程序编号n从0开始自动向上生成，如图4-109所示。

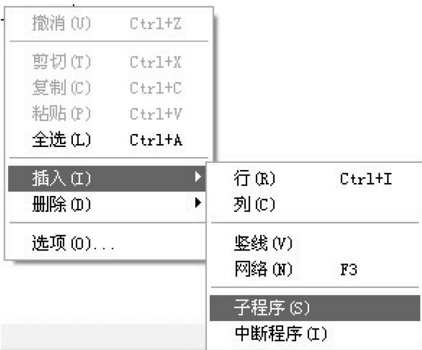


图4-109　插入子程序

【例4-59】编写一段程序将VW0中的数转换成实数存入VD2中。

【解】

S7-200中没有直接将整数转换成实数指令，因此先编写子程序如图4-110所示，注意先要设置参数表。再在主程序中调用子程序，如图4-111所示。

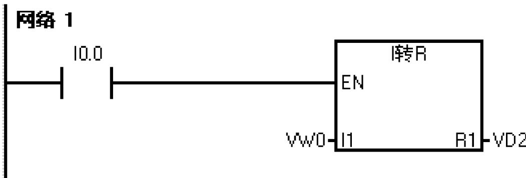


图4-110　主程序

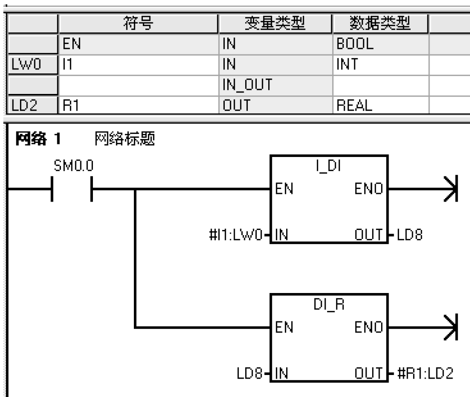


图4-111　子程序

4.5.2 中断调用

中断是计算机特有的工作方式，即在主程序的执行过程中中断主程序，而执行中断子程序。中断子程序是为某些特定的控制功能而设定的。与子程序不同，中断是为随机发生的且必须立即响应的时间安排的，其响应时间应小于机器周期。引发中断的信号称为中断源，S7－200 有 34 个中断源，见表 4-45。

表 4-45　S7－200 的 34 个中断源

序号	中 断 描 述	CPU 221 CPU 222	CPU 224	CPU 226 224XP	序号	中 断 描 述	CPU 221 CPU 222	CPU 224	CPU 226 224 XP
0	上升沿，I0. 0	✓	✓	✓	17	HSC2 输入方向改变		✓	✓
1	下降沿，I0. 0	✓	✓	✓	18	HSC2 外部复位		✓	✓
2	上升沿，I0. 1	✓	✓	✓	19	PTO 0 完成中断	✓	✓	✓
3	下降沿，I0. 1	✓	✓	✓	20	PTO 1 完成中断	✓	✓	✓
4	上升沿，I0. 2	✓	✓	✓	21	定时器 T32 CT = PT 中断	✓	✓	✓
5	下降沿，I0. 2	✓	✓	✓	22	定时器 T96 CT = PT 中断	✓	✓	✓
6	上升沿，I0. 3	✓	✓	✓	23	端口 0：接收信息完成	✓	✓	✓
7	下降沿，I0. 3	✓	✓	✓	24	端口 1：接收信息完成			✓
8	端口 0：接收字符	✓	✓	✓	25	端口 1：接收字符			✓
9	端口 0：发送完成	✓	✓	✓	26	端口 1：发送完成			✓
10	定时中断 0 SMB34	✓	✓	✓	27	HSC0 输入方向改变	✓	✓	✓
11	定时中断 1 SMB35	✓	✓	✓	28	HSC0 外部复位	✓	✓	✓
12	HSC0 CV = PV	✓	✓	✓	29	HSC4 CV = PV	✓	✓	✓
13	HSC1 CV = PV		✓	✓	30	HSC4 输入方向改变	✓	✓	✓
14	HSC1 输入方向改变		✓	✓	31	HSC4 外部复位	✓	✓	✓
15	HSC 外部复位		✓	✓	32	HSC3 CV = PV	✓	✓	✓
16	HSC2 CV = PV		✓	✓	33	HSC5 CV = PV	✓	✓	✓

注：“✓”表明对应的 CPU 有相应的中断功能。

1. 中断的分类

S7－200 的 34 个中断源可分为三大类，即 I/O 口中断、通信口中断和时基中断。

（1）I/O 口中断

I/O 口中断包括上升沿和下降沿中断、高速计数器中断和脉冲串输出中断。S7－200 可以利用 I0. 0 ~ I0. 3 都有上升沿和下降沿这一特性产生中断事件。

（2）通信口中断

通信口中断包括端口 0（Port0）和端口 1（Port1）接收和发送中断。PLC 的串行通信口可由程序控制，这种模式称为自由口通信模式，在这种模式下通信，接收和发送中断可以简化程序。

（3）时基中断

时基中断包括定时中断及定时器 T32/96 中断。定时中断可以反复执行，定时中断是非常有用的。

2. 中断指令

中断指令共有 6 条，包括中断连接、中断分离、清除中断事件、中断禁止、中断允许和中断条件返回，见表 4-46。

表 4-46　中断指令

LAD	STL	功　能
ATCH EN ENO INT EVNT	ATCH，INT，EVNT	中断连接
DTCH EN ENO EVNT	DTCH，EVNT	中断分离
CLR_EVNT EN ENO EVNT	CENT，EVNT	清除中断事件
—(DISI)	DISI	中断禁止
—(ENI)	ENI	中断允许
—(RETI)	CRETI	中断条件返回

【例 4-60】 用中断指令编写程序，要求实现每隔 100ms VD0 中的数值增加 1。

【解】

程序如图 4-112 所示，其中用到了中断。

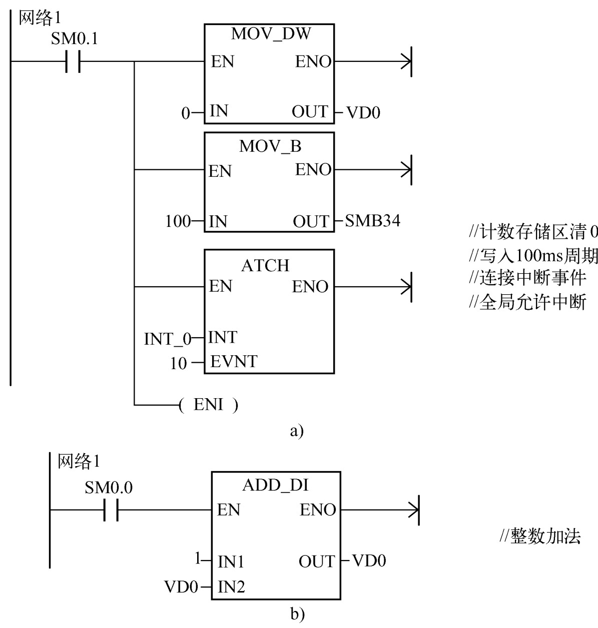


图 4-112　程序

a）主程序　b）中断程序 INT_0

注意：

① 一个事件只能连接一个中断程序，而多个中断事件可以调用同一个中断程序，但一个中断事件不可能在同一时间建立多个中断程序。

② 在中断子程序中不能使用 DISI、ENI、HDFE、FOR/NEXT 和 END 等指令。

③ 程序中有多个中断子程序时，要分别编号。在建立中断程序时，系统会自动编号，也可以更改编号。建立子程序的方法有三种，最简单的方法是在程序编辑器中的空白处单击鼠标右键，再单击快捷菜单“插入”下的子菜单“中断程序”即可，如图 4-113 所示。

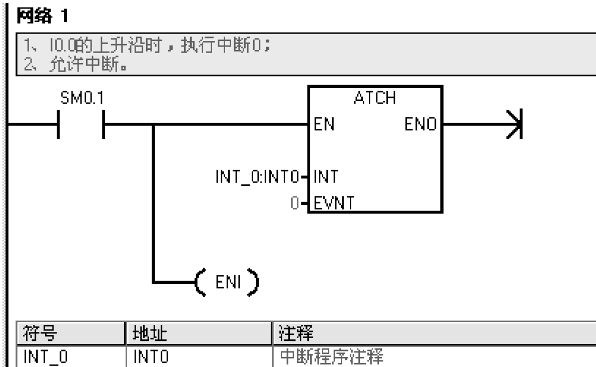


图 4-113 插入中断程序

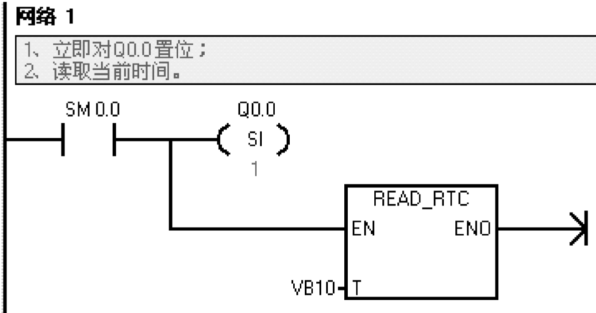
【例 4-61】 记录一台设备损坏（设备损坏时接通 I0.0）的时间，请用 PLC 实现此功能。

【解】

程序如图 4-114 所示。



a)



b)

图 4-114 中断指令程序

a）主程序 b）中断程序

【例 4-62】在 I0.0 的上升沿，通过中断使 Q0.0 立即置位，在 I0.0 的下降沿，通过中断使 Q0.0 立即复位。

【解】

梯形图如图 4-115 所示。

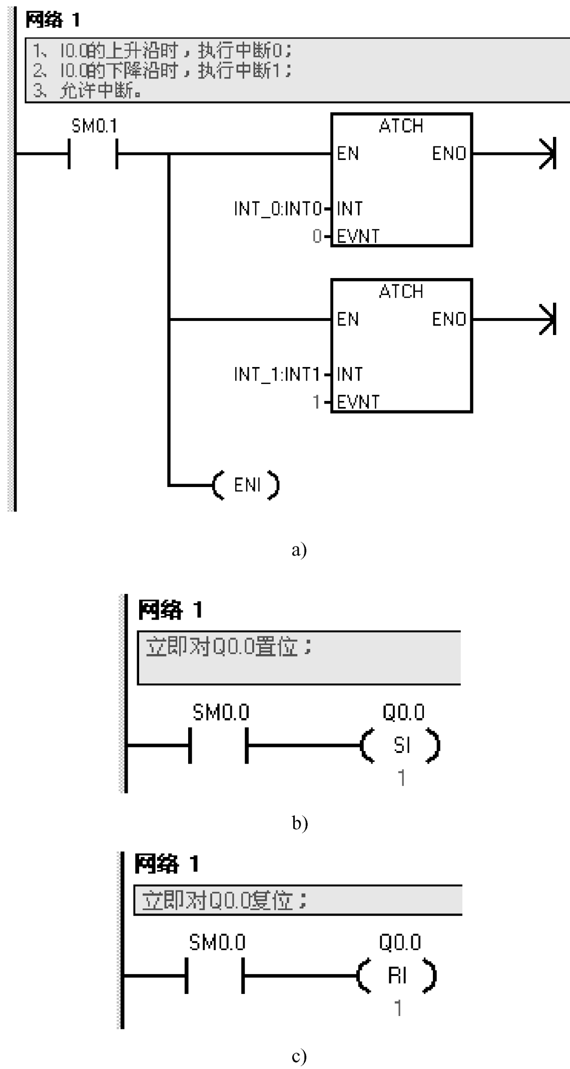


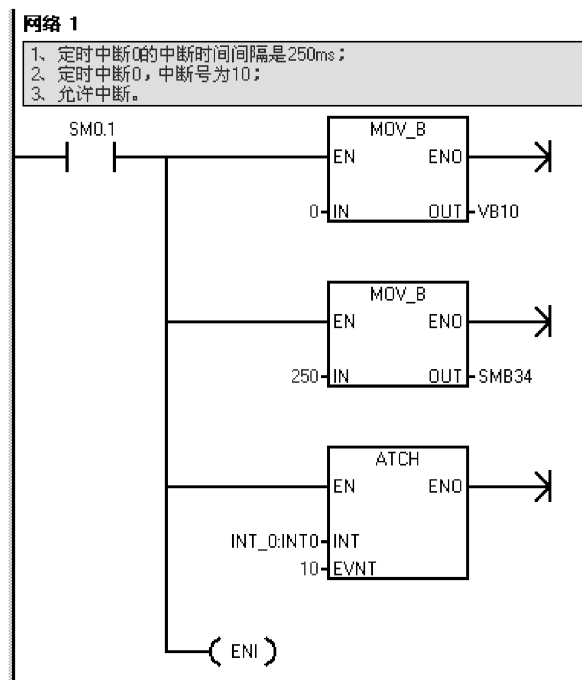
图 4-115　梯形图

a）主程序　b）中断程序 INT_0　c）中断程序 INT_1

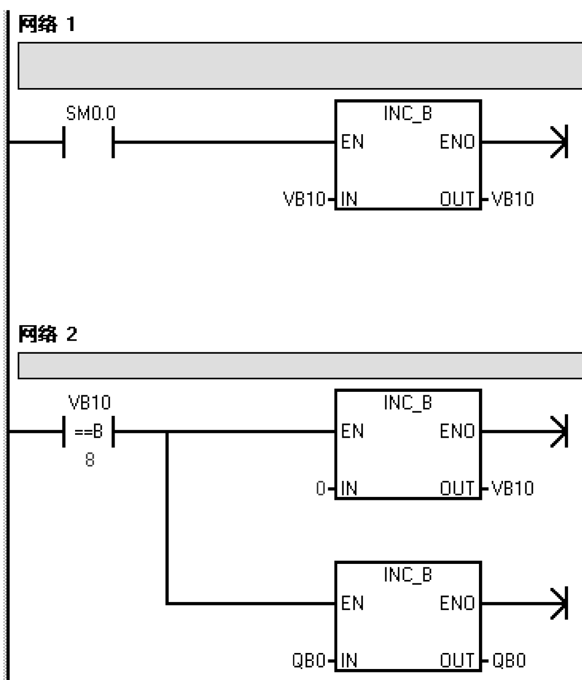
【例 4-63】用定时中断 0，设计一段程序，实现周期为 2 s 的精确定时。

【解】

SMB34 是存放定时中断 0 的定时长短的特殊寄存器，其最大定时时间是 255 ms，2 s 就是 8 次 250 ms 的延时。图 4-116 所示为梯形图。



a)



b)

图 4-116　梯形图

a）主程序　b）子程序

4.5.3 跳转指令

跳转指令（JMP）和跳转地址标号（LBL）配合实现程序的跳转。使能端输入有效时，程序跳转到指定标号 n 处（同一程序内），跳转标号 n＝0～255；使能端输入无效时，程序顺序执行。跳转指令格式见表 4-47。

表 4-47　跳转指令格式

LAD	STL	功　　能
——(JMP) n	JMP　　n	跳转指令
n [LBL]	LBL　　n	跳转标号

跳转指令的使用要注意如下几点：

① 允许多条跳转指令使用同一标号，但不允许一个跳转指令对应两个标号，同一个指令中不能有两个相同的标号。

② 跳转指令具有程序选择功能，类似于 BASIC 语言的 GOTO 指令。

③ 主程序、子程序和中断服务程序中都可以使用跳转指令，SCR 网络中也可以使用跳转指令。

④ 若跳转指令中使用上升沿或者下降沿脉冲指令时，跳转只执行一个周期，但若使用 SM0.0 作为跳转条件，跳转则称为无条件跳转。

跳转指令程序示例如图 4-117 所示。

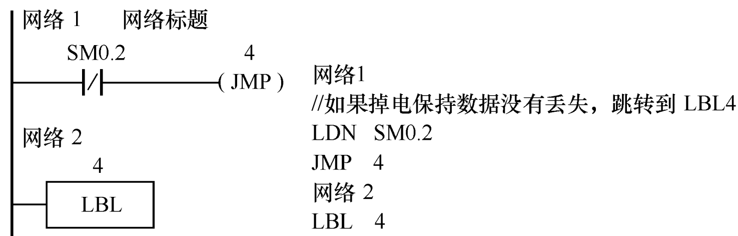


图 4-117　跳转指令程序示例

4.5.4 循环指令

循环指令（FOR－NEXT）用于一段程序的重复循环执行，由 FOR 指令和 NEXT 指令构成程序的循环体，FOR 标记循环的开始，NEXT 为循环体的结束指令，见表 4-48。FOR 指令为指令盒格式，主要参数有使能输入 EN、当前值计数器 INDX、循环次数初始值 INIT 和循环计数终值 FINAL。

表 4-48　循环指令格式

LAD	STL	功　能
FOR（EN ENO，INDX，INIT，FINAL）	FOR　IN1，IN2，IN3	循环开始
—(NEXT)	NEXT	循环返回

当使能输入 EN 有效时，循环体开始执行，执行到 NEXT 指令时返回。每执行一次循环体，当前计数器 INDX 增 1，达到终值 FINAL 时，循环结束。FINAL 为 10，使能有效时，执行循环体，同时 INDX 从 1 开始计数，每执行一次循环体，INDX 当前值加 1，执行到 10 次时，当前值也计到 11，循环结束。

使能输入无效时，循环体程序不执行。FOR 指令和 NEXT 指令必须成对使用，循环可以嵌套，最多为 8 层。循环指令应用示例如图 4-118 所示。

【例 4-64】 在如图 4-118 所示程序中，单击 2 次按钮 I0.0 后，VW0 和 VB10 中的数值是多少？

【解】

单击 2 次按钮，执行 2 次循环程序，VB10 执行 20 次加 1 运算，所以 VB10 结果为 20。执行 1 次或者 2 次循环程序，VW0 中的值都为 11。

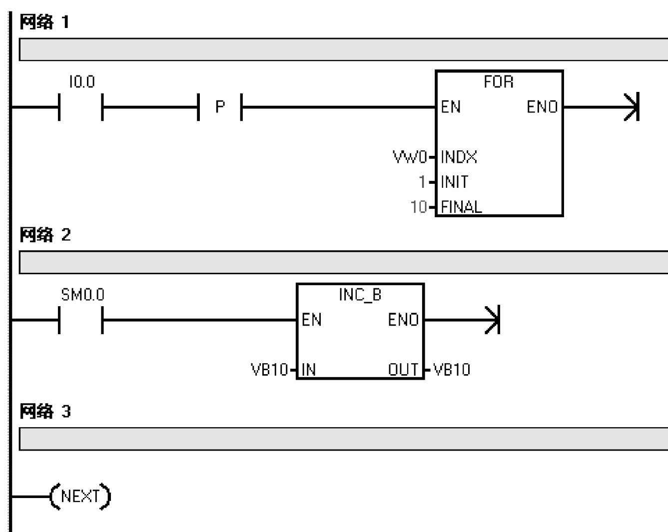


图 4-118　循环指令应用示例

【关键点】 I0.0 后面要有一个上升沿“P”（或者“N”），否则按下一次按钮，运行 INC 指令的次数是不确定数，一般远多于程序中的 10 次。

4.5.5　暂停指令

使能端输入有效时，立即停止程序的执行。指令执行的结果是，CPU 的工作方式由

RUN 切换到 STOP 方式。暂停指令（STOP）格式见表 4-49。

表 4-49　暂停指令格式

LAD	STL	功　能
——(STOP)	STOP	暂停程序执行

4.5.6　结束指令

结束指令（END/MEND）：直接连在左侧母线时，为无条件结束指令（MEND）；不连在左侧母线时，为条件结束指令。结束指令格式见表 4-50。

表 4-50　结束指令格式

LAD	STL	功　能
——(END)	END	条件结束指令
├——(END)	MEND	无条件结束指令

条件结束指令在使能端输入有效时，终止用户程序的执行，返回主程序的第一条指令行（循环扫描方式）。

无条件结束指令执行时（指令直接连在左侧母线上，无使能输入），立即终止用户程序的执行，返回主程序的第一条指令行。

结束指令只能在主程序中使用，不能在子程序和中断服务程序中使用。

STEP 7 - Micro/WIN 编程软件会在主程序的结尾处自动生成无条件结束指令，用户不用输入无条件结束指令，否则编译出错。

4.5.7　指针

间接寻址是指用指针来访问存储区数据。指针以双字的形式存储其他存储区的地址。只能用 V 存储器、L 存储器或者累加器寄存器（AC1、AC2、AC3）作为指针。要建立一个指针，必须以双字的形式，将需要间接寻址的存储器地址移动到指针中。指针也可以为子程序传递参数。

S7 - 200 允许指针访问以下存储区：I、Q、V、M、S、AI、AQ、SM、T（仅限于当前值）和 C（仅限于当前值）。无法用间接寻址的方式访问位地址，也不能访问 HC 或者 L 存储区。

要使用间接寻址，应该用“&”符号加上要访问的存储区地址来建立一个指针。指令的输入操作数应该以“&”符号开头来表明是存储区的地址，而不是其内容将移动到指令的输出操作数（指针）中。

当指令中的操作数是指针时，应该在操作数前面加上“*”号。如图 4-119 所示，输入 *AC1 指定 AC1 是一个指针，MOVW 指令决定了指针指向的是一个字长的数据。在本例中，存储在 VB200 和 VB201 中。

例如：MOVD &VB200，AC1，其含义是将 VB200 的地址（VB200 的起始地址）作为指针存入 AC1 中。MOVW *AC1，AC0，其含义是将 AC1 指向的字送到 AC0 中去。

图 4-119 中，输入 *AC1 指定 AC1 是一个指针，MOVW 指令决定了指针指向的是一个字长的数据。在本例中，存储在 VB200 和 VB201 中。

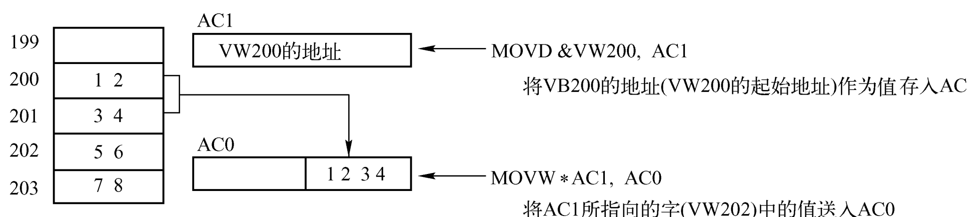


图 4-119　指针的使用

【例 4-65】 将非线性的表格存放在 VW0 开始的 100 字中，表格偏移量（表格中字的序号，第一个字序号为 0）在 VD200 中，用间隔寻址将表格中相对于偏移量的数据传送到 VW210 中去。

【解】

程序如图 4-120 所示。

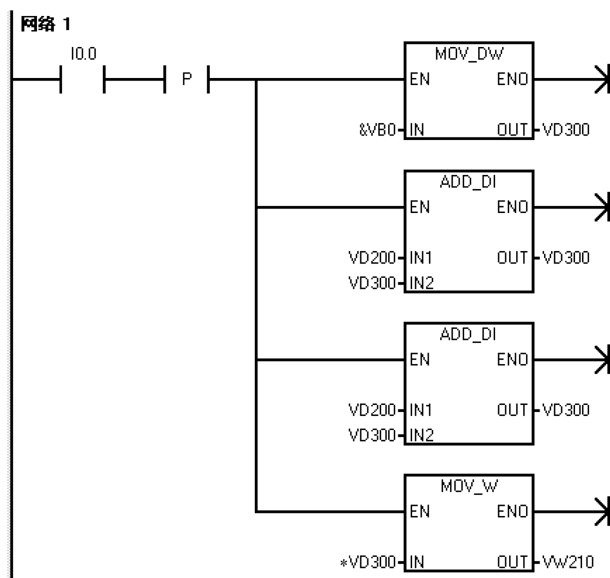


图 4-120　程序

4.5.8　顺控继电器指令

顺控继电器指令又称 SCR，S7-200 系列 PLC 有三条顺控继电器指令，指令格式见表 4-51。

表 4-51　顺控继电器指令格式

LAD	STL	功　能
n ├[SCR]	LSCR，n	装载顺控继电器指令，将 S 位的值装载到 SCR 和逻辑堆栈中，实际是步开始指令
n —(SCRT)	SCRT，n	使当前激活的 S 位复位，使下一个将要执行的网络 S 置位，实际上是步转移指令
├(SCRE)	SCRE	退出一个激活的网络，实际上是步结束指令

顺控继电器指令编程时应注意：

① 不能把 S 位用于不同的程序中。例如 S0.2 已经在主程序中使用了，就不能在子程序

中使用。

② 顺控继电器指令 SCR 只对状态元件 S 有效。

③ 不能在 SCR 段中使用 FOR、NEXT 和 END 指令。

④ 在 SCR 之间不能有跳入和跳出，也就是不能使用 JMP 和 LBL 指令。但注意，可以在 SCR 网络附近和 SCR 网络内使用跳转指令。

【例 4-66】 用 PLC 控制一盏灯亮 0.3 s 后熄灭，再控制另一盏灯亮 0.3 s 后熄灭，周而复始重复以上过程，要求根据图 4-121 所示的功能图，使用顺控继电器指令编写程序。

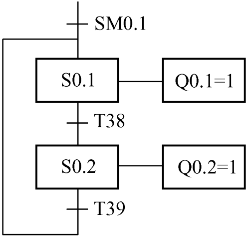
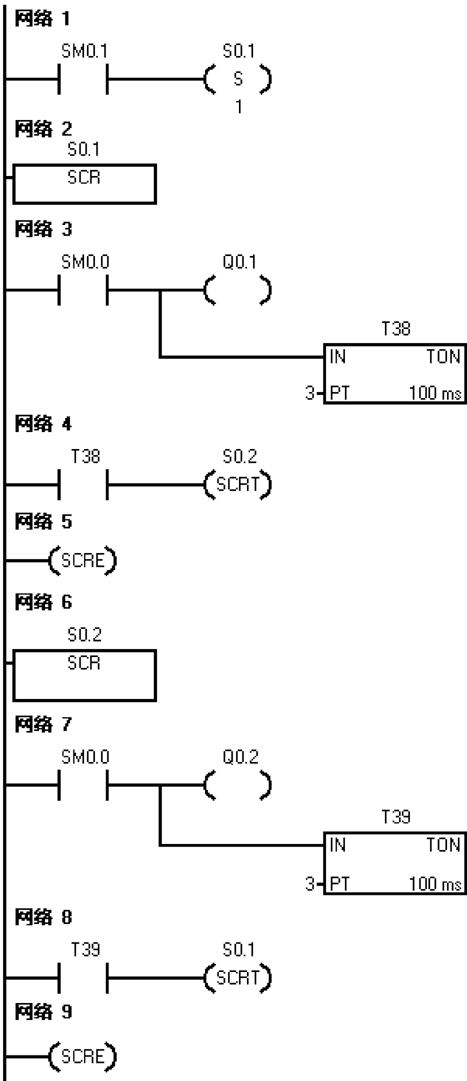


图 4-121　功能图

【解】

在已知功能图的情况下，用顺控指令编写程序是很容易的，程序如图 4-122 所示。




```
网络1
LD     SM0.1
S      S0.1, 1
网络2
LSCR   S0.1
网络3
LD     SM0.1
=      Q0.1
TON    T38, 3
网络4
LD     T38
SCRT   S0.2
网络5
SCRE
网络6
LSCR   S0.2
网络7
LD     SM0.0
=      Q0.2
TON    T39, 3
网络8
LD     T39
SCRT   S0.1
网络9
SCRE
```

图 4-122　梯形图

4.5.9 程序控制指令的应用

【例 4-67】 某系统测量温度，当温度超过一定数值（保存在 VW10 中）时，报警灯以 1s 为周期闪光，警铃鸣叫，使用 S7－200 PLC 和模块 EM 231，编写此程序。

【解】

温度是一个变化较慢的量，可每隔 100 ms 从模块 EM 231 的通道 0 中采样 1 次。程序如图 4-123 所示。

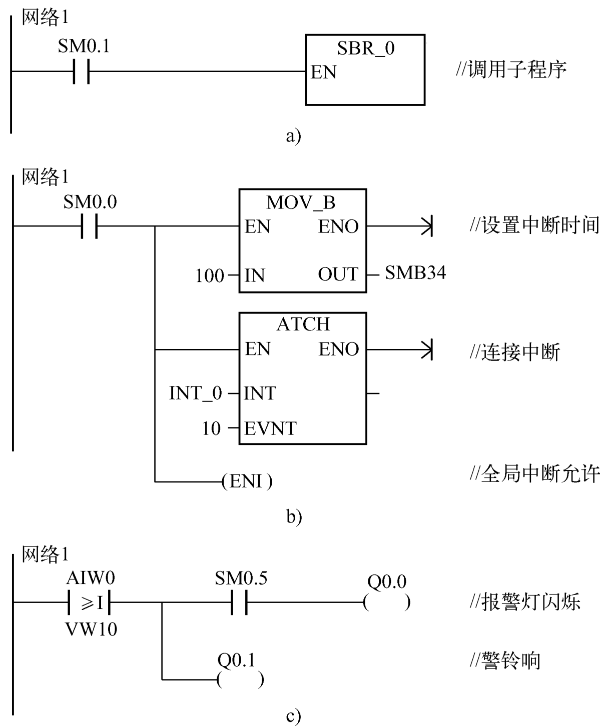


图 4-123　程序

小　　结

（1）数据类型。

（2）重点掌握 S7－200 可编程序控制器的基本逻辑指令、功能指令和顺序继电器指令的格式。其中，基本指令是重点，而功能指令是难点。

（3）重点掌握 S7－200 可编程序控制器 I/O 接线图画法。学会查阅 PLC 用户手册。

（4）本章的例子多达 67 个，这对学习很有帮助，请读者注意理解。

习　　题

1. 写出图 4-124 所示梯形图所对应的语句表指令。

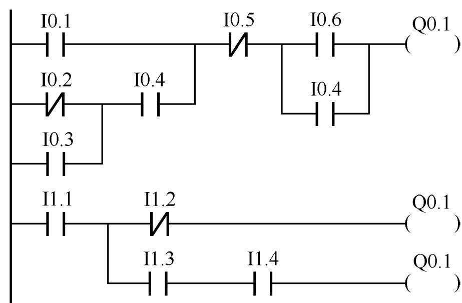


图 4-124　习题 1 附图

2. 根据下列语句表程序，画出梯形图。

```
LD    I0.0
AN    I0.1
LD    I0.2
A     I0.3
O     I0.4
A     I0.5
OLD
LPS
A     I0.6
=     Q0.1
LPP
A     I0.7
=     Q0.2
A     I1.1
=     Q0.3
```

3. 3 台电动机相隔 5 s 起动，各运行 20 s，循环往复。使用传送指令和比较指令完成控制要求。

4. 用 PLC 设计一个闹钟，每天早上 6: 00 响铃。

5. 用 PLC 的置位、复位指令实现彩灯的自动控制。控制过程为：按下起动按钮，第一组花样绿灯亮；10 s 后第二组花样蓝灯亮；20 s 后第三组花样红灯亮；30 s 后返回第一组花样绿灯亮。如此循环，并且仅在第三组花样红灯亮后方可停止循环。

6. 如图 4-125 所示为一台电动机起动的工作时序图，试画出梯形图。

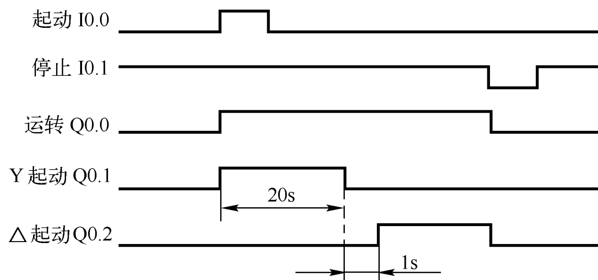


图 4-125　习题 6 附图

7. 用3个开关（I0.1、I0.2、I0.3）控制一盏灯 Q1.0，当3个开关全通或者全断时灯亮，其他情况灯灭。（提示：使用比较指令。）

8. 用移位指令构成移位寄存器，实现广告牌字的闪耀控制。用 HL1 ~ HL4 四只灯分别照亮“欢迎光临”四个字，其控制要求见表4-52，每步间隔1 s。

表4-52　广告牌字闪耀流程

流　　程	1	2	3	4	5	6	7	8
HL1	✓				✓		✓	
HL2		✓			✓		✓	
HL3			✓		✓		✓	
HL4				✓	✓		✓	

9. 运用算术运算指令完成算式[（100+200）×10]/3 的运算，并画出梯形图。

10. 编写一段程序，将 VB100 开始的50个字的数据传送到 VB1000 开始的存储区。

11. 编写将 VW100 的高、低字节内容互换并将结果送入定时器 T37 作为定时器预置值的网络。

12. 某系统上有1个 S7-226 CPU、2个 EM 221 模块和3个 EM 223 模块，计算由 CPU 226 供电，电源是否足够？

13. 以下哪些表达有错误？请改正。

AQW3、8#11、10#22、16#FF、16#FFH、2#110、2#21

14. 现有3台电动机 M_1、M_2、M_3，要求按下起动按钮 I0.0 后，电动机按顺序起动（M_1 起动，接着 M_2 起动，最后 M_3 起动），按下停止按钮 I0.1 后，电动机按顺序停止（M_3 先停止，接着 M_2 停止，最后 M_1 停止）。试设计其梯形图并写出指令表。

15. 如图4-126所示，若传送带上20s内无产品通过则报警，并接通 Q0.0。试画出梯形图并写出指令表。

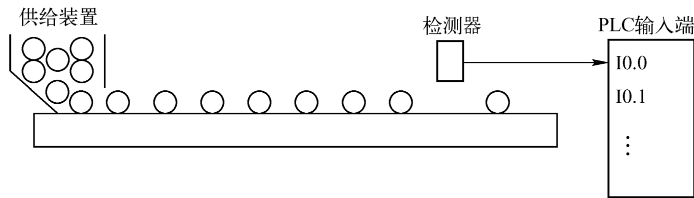


图4-126　习题15附图

16. 如图4-127所示为两组带机组成的原料运输自动化系统，该自动化系统的起动顺序为：盛料斗 D 中无料，先起动带机 C，5 s 后再起动带机 B，经过7 s 后再打开电磁阀 YV，该自动化系统停机的顺序恰好与起动顺序相反。试完成梯形图设计。

17. 试用 DECO 指令实现某喷水池花式喷水控制。控制流程要求为第一组喷嘴喷水4 s，第二组喷嘴喷水2 s，两组喷嘴同时喷水2 s，都停止喷水1 s，重复以上过程。

18. 编写一段检测上升沿变化的程序。每当 I0.1 接通一次，VB0 的数值增加1，如果计数达到18时，Q0.1 接通，用 I0.2 使 Q0.1 复位。

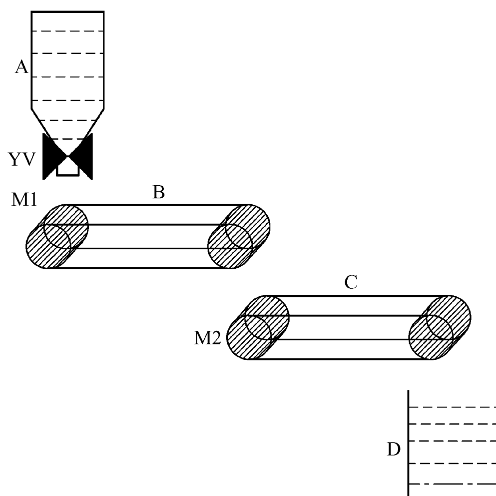


图 4-127　习题 16 附图

19. 第 1 次按下按钮指示灯亮，第 2 次按下按钮指示灯闪亮，第 3 次按下按钮指示灯灭，如此循环，试编写其 PLC 控制的 LAD 程序。

20. 用一个按钮控制 2 盏灯，第 1 次按下时第 1 盏灯亮，第 2 盏灯灭；第 2 次按下时第 1 盏灯灭，第 2 盏灯亮；第 3 次按下时 2 盏灯都灭。

21. 编写 PLC 控制程序，使 Q0.0 输出周期为 5s，占空比为 20% 的连续脉冲信号。

22. 根据如图 4-128 所示的时序图画出梯形图。

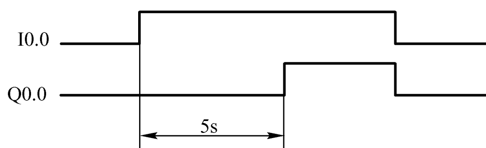


图 4-128　习题 22 附图

23. 设计故障显示程序，若故障显示信号 I0.0 为 1 状态，Q1.0 控制的指示灯以 2 Hz 的频率闪烁。操作人员按复位按钮 I1.0 后，如果故障已经消失，指示灯熄灭。如果没有消失，指示灯转为常亮，直至故障消失。

24. 用寄存器移位指令 SHRB 设计一梯形图，要求如下：

（1）当脉冲上升沿到来时，SHRB 指令将 QB1 字节右移一位，时基脉冲由 SM0.5（1 s）产生。

（2）要用输出继电器作显示。

25. 用 PLC 设计一个闹钟，每天早上 6:00 响铃，人工复位。

第 5 章　逻辑控制编程的编写方法

本章介绍顺序功能图的画法、梯形图的禁忌以及如何根据顺序功能图用基本指令、功能指令、复位/置位指令和顺控指令四种方法编写逻辑控制的梯形图。

5.1　顺序功能图

5.1.1　顺序功能图的画法

顺序功能图（Sequential function chart，SFC）又叫做状态转移图，它是描述控制系统的控制过程、功能和特性的一种图形，同时也是设计 PLC 顺序控制程序的一种有力工具。它具有简单、直观等特点，不涉及控制功能的具体技术，是一种通用的语言，是 IEC（国际电工委员会）首选的编程语言，近年来在 PLC 的编程中已经得到了普及与推广。在 IEC848 中称顺序功能图，在我国国家标准 GB6988—1986 中称功能表图。西门子称为图形编程语言 S7－Graph 和 S7－HiGraph。

顺序功能图是设计 PLC 顺序控制程序的一种工具，适合于系统规模较大，程序关系较复杂的场合，特别适合于对顺序操作的控制。在编写复杂的顺序控制程序时，采用 S7－Graph 和 S7－HiGraph 比梯形图更加直观。

顺序功能图的基本思想是：设计者按照生产要求，将被控设备的一个工作周期划分成若干个工作阶段（简称“步”），并明确表示每一步要执行的输出，“步”与“步”之间通过指定的条件进行转换，在程序中，只要通过正确连接进行“步”与“步”之间的转换，就可以完成被控设备的全部动作。

PLC 执行顺序功能图程序的基本过程是：根据转换条件选择工作“步”，进行“步”的逻辑处理。组成顺序功能图程序的基本要素是步、转换条件和有向连线，如图 5-1 所示。

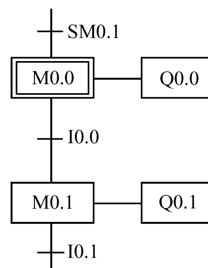


图 5-1　顺序功能图

（1）步

一个顺序控制过程可分为若干个阶段，也称为步或状态。系统初始状态对应的步称为初始步，初始步一般用双线框表示。在每一步中施控系统要发出某些“命令”，而被控系统要完成某些“动作”，“命令”和“动作”都称为动作。当系统处于某一工作阶段时，则该步处于激活状态，称为活动步。

（2）转换条件

使系统由当前步进入下一步的信号称为转换条件。顺序控制设计法用转换条件控制代表各步的编程元件，让它们的状态按一定的顺序变化，然后用代表各步的编程元件去控制输出。不同状态的“转换条件”可以不同，也可以相同，当“转换条件”各不相同时，在顺

序功能图程序中每次只能选择其中一种工作状态（称为“选择分支”），当“转换条件”都相同时，在顺序功能图程序中每次可以选择多个工作状态（称为“选择并行分支”）。只有满足条件状态，才能进行逻辑处理与输出。因此，“转换条件”是顺序功能图程序选择工作状态（步）的“开关”。

（3）有向连线

步与步之间的连接线就是“有向连线”，“有向连线”决定了状态的转换方向与转换途径。在有向连线上有短线，表示转换条件。当条件满足时，转换得以实现，即上一步的动作结束而下一步的动作开始，因而不会出现动作重叠。步与步之间必须要有转换条件。

图 5-1 中的双框为初始步，M0.0、M0.1 是步名，I0.0、I0.1 为转换条件，Q0.0、Q0.1 为动作。当 M0.0 有效时，输出指令驱动 Q0.0。步与步之间的连线称为有向连线，有向连线的箭头省略未画。

（4）顺序功能图的结构分类

根据步与步之间的进展情况，顺序功能图分为以下 3 种结构。

1）单一顺序。单一顺序动作是一个接一个的完成，完成每步只连接一个转移，每个转移只连接一个步，分别如图 5-3 和图 5-4 所示。功能图和梯形图是一一对应关系，以下用“起保停电路”来讲解这种对应关系。

为了便于将顺序功能图转换为梯形图，采用代表各步的编程元件的地址（比如 M0.2）作为步的代号，并用编程元件的地址来标注转换条件和各步的动作和命令，当某步对应的编程元件置 1，代表该步处于活动状态。

① 起保停电路对应的布尔代数式。标准的起保停梯形图如图 5-2 所示，图中 I0.0 为 M0.2 的起动条件，当 I0.0 置 1，M0.2 得电；I0.1 为 M0.2 的停止条件，当 I0.1 置 1，M0.2 断电；M0.2 的辅助触点为 M0.2 的保持条件。该梯形图对应的布尔代数式为：

$$M0.2=(I0.0+M0.2)\cdot\overline{I0.1}$$

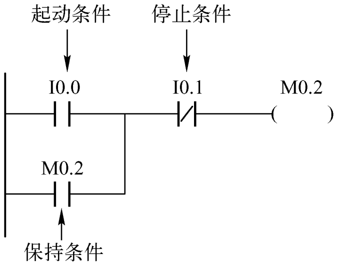


图 5-2　标准的“起保停”梯形图

② 顺序控制梯形图储存位对应的布尔代数式。功能图如图 5-3a 所示，M0.1 转换为活动步的条件是 M0.1 步的前一步是活动步，相应的转换条件（I0.0）得到满足，即 M0.1 的起动条件为 M0.0 · I0.0。当 M0.2 转换为活动步后，M0.1 转换为不活动步，因此，M0.2 可以看成 M0.1 的停止条件。由于大部分转换条件都是瞬时信

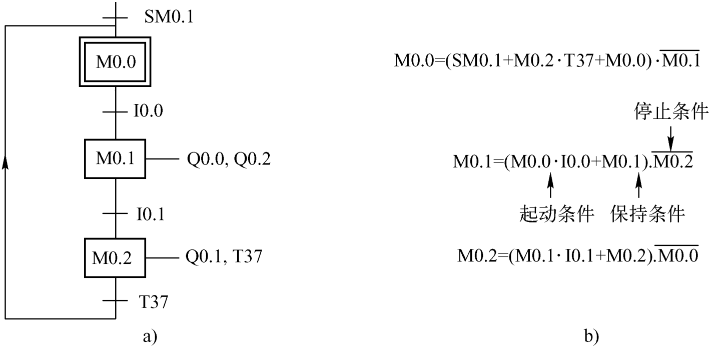


图 5-3　顺序功能图和对应的布尔代数式

号，即信号持续的时间比它激活的后续步的时间短，因此应当使用有记忆功能的电路控制代表步的储存位。在这种情况下，注意到，起动条件、停止条件和保持条件就全部都有了，就可以用起保停方法来设计顺序功能图的布尔代数式和梯形图。顺序控制功能图中储存位对应的布尔代数式如图 5-3b 所示，参照图 5-2 所示的标准“起保停”梯形图，就可以轻松地将图 5-3 所示的顺序功能图转换为如图 5-4 所示的梯形图。

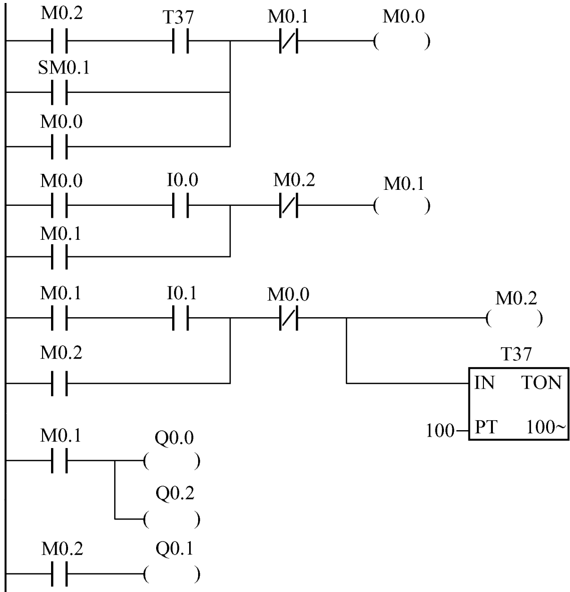


图 5-4　梯形图

2）选择顺序。选择顺序是指某一步后有若干个单一顺序等待选择，称为分支，一般只允许选择进入一个顺序，转换条件只能标在水平线之下。选择顺序的结束称为合并，用一条水平线表示，水平线以下不允许有转换条件，如图 5-5 所示。

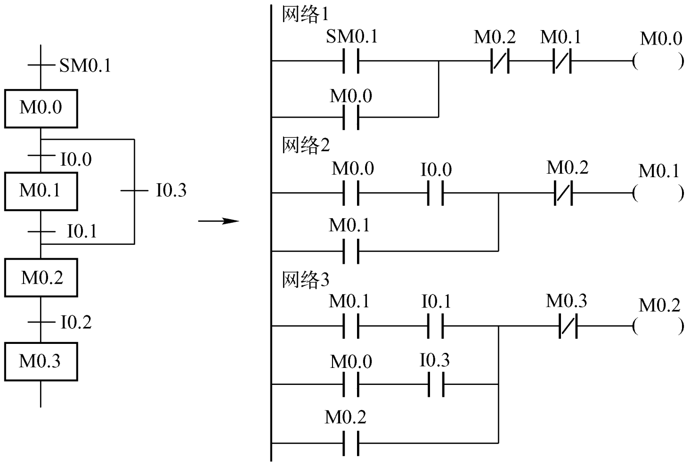


图 5-5　选择顺序

3）并行顺序。并行顺序是指在某一转换条件下同时起动若干个顺序，也就是说转换条件的实现导致几个分支同时激活。并行顺序的开始和结束都用双水平线表示，如图 5-6 所示。

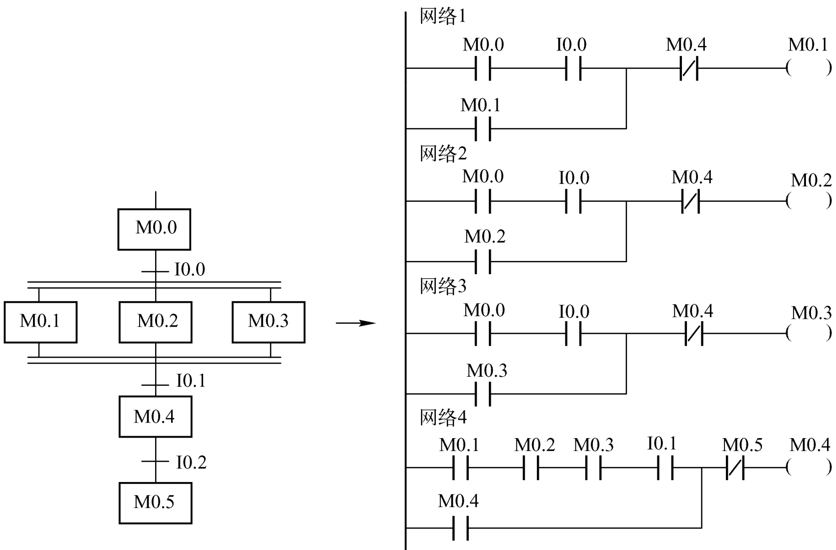


图 5-6　并行顺序

4）选择序列和并行序列的综合。如图 5-7 所示，步 M0.0 之后有一个选择序列的分支，设 M0.0 为活动步，当它的后续步 M0.1 或 M0.2 变为活动步时，M0.0 变为不活动步，即 M0.0 为 0 状态，所以应将 M0.1 和 M0.2 的常闭触点与 M0.0 的线圈串联。

步 M0.2 之前有一个选择序列合并，当步 M0.1 为活动步（即 M0.1 为 1 状态），并且转换条件 I0.1 满足，或者步 M0.0 为活动步，并且转换条件 I0.2 满足，步 M0.2 变为活动步，所以该步的存储器 M0.2 的起保停电路的起动条件为 M0.1 · I0.1 + M0.0 · I0.2，对应的起动电路由两条并联支路组成。

步 M0.2 之后有一个并行序列分支，当步 M0.2 是活动步并且转换条件 I0.3 满足时，步 M0.3 和步 M0.5 同时变成活动步，这时用 M0.2 和 I0.3 常开触点组成的串联电路，分别作为 M0.3 和 M0.5 的起动电路来实现，与此同时，步 M0.2 变为不活动步。

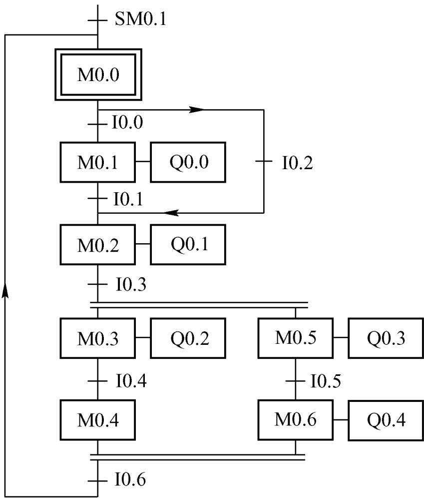


图 5-7　选择序列和并行序列功能图

步 M0.0 之前有一个并行序列的合并，该转换实现的条件是所有的前级步（即 M0.4 和 M0.6）都是活动步和转换条件 I0.6 满足。由此可知，应将 M0.4、M0.6 和 I0.6 的常开触点串联，作为控制 M0.0 的起保停电路的起动电路。图 5-7 所示的功能图对应的梯形图如图 5-8 所示。

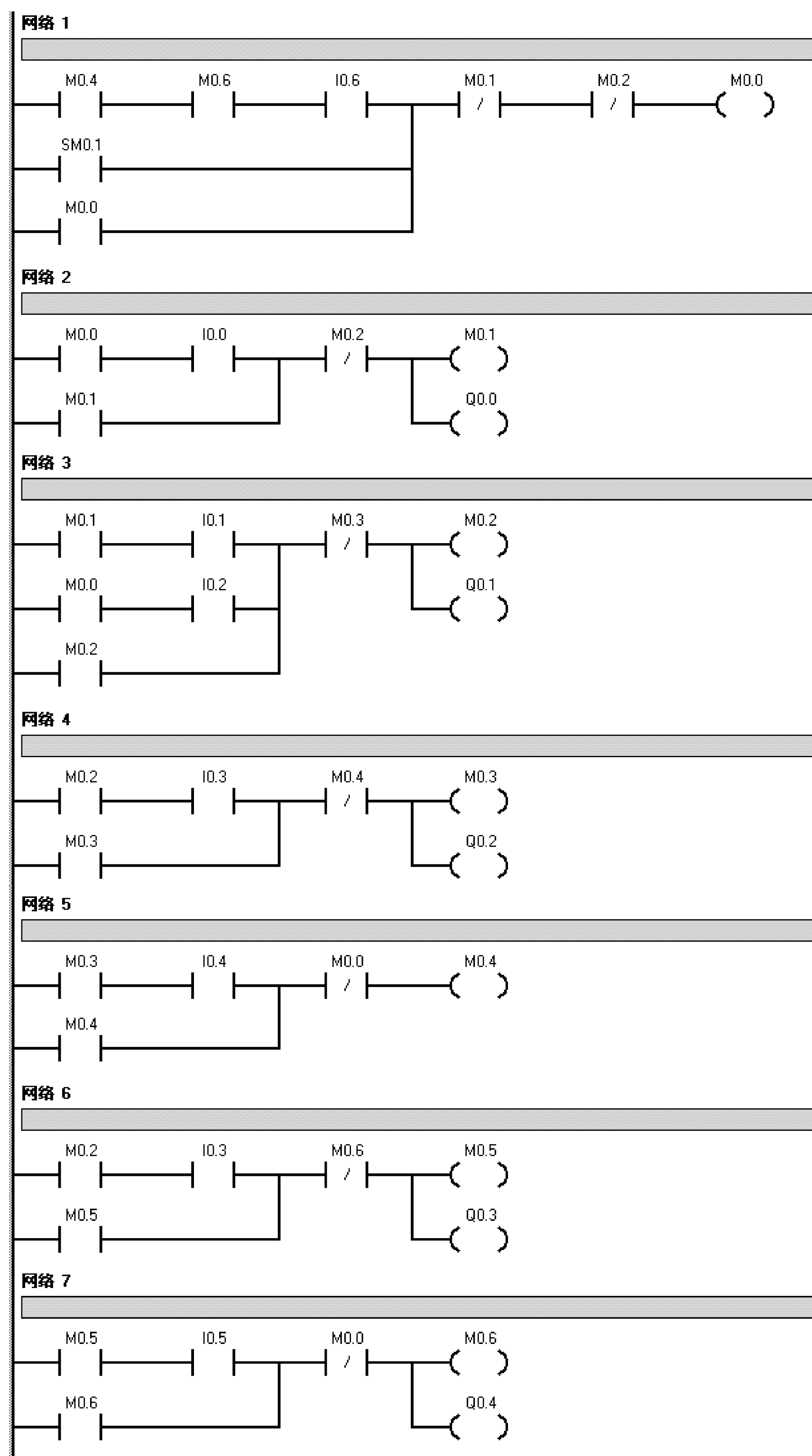

网络 1
M0.4
M0.6
I0.6
M0.1
M0.2
M0.0
SM0.1
M0.0
网络 2
M0.0
I0.0
M0.2
M0.1
M0.1
Q0.0
网络 3
M0.1
I0.1
M0.3
M0.2
M0.0
I0.2
Q0.1
M0.2
网络 4
M0.2
I0.3
M0.4
M0.3
M0.3
Q0.2
网络 5
M0.3
I0.4
M0.0
M0.4
M0.4
网络 6
M0.2
I0.3
M0.6
M0.5
M0.5
Q0.3
网络 7
M0.5
I0.5
M0.0
M0.6
M0.6
Q0.4

图 5-8　梯形图

（5）顺序功能图设计的注意点

1）状态之间要有转换条件，如图 5-9 中，状态之间缺少“转换条件”是不正确的，应改成如图 5-10 所示的顺序功能图。必要时转换条件可以简化，应将图 5-11 简化成图 5-12。

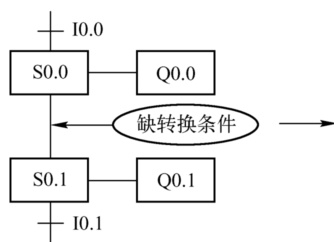


图 5-9　错误的顺序功能图

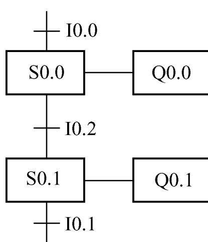


图 5-10　正确的顺序功能图

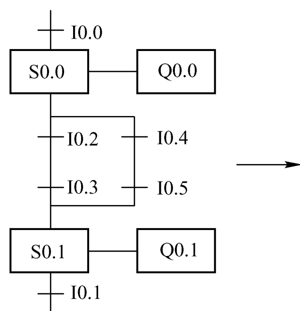


图 5-11　简化前的顺序功能图

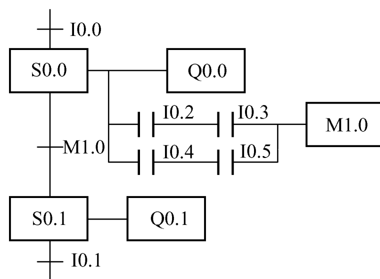


图 5-12　简化后的顺序功能图

2）转换条件之间不能有分支，例如，图 5-13 应该改成如图 5-14 所示的合并后的顺序功能图，合并转换条件。

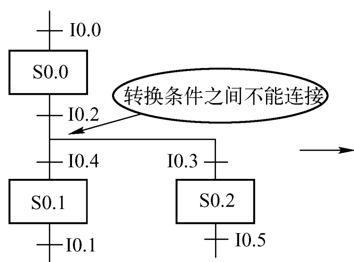


图 5-13　错误的顺序功能图

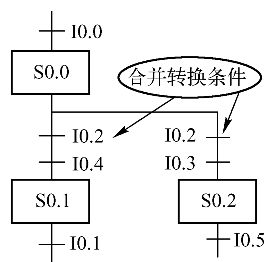


图 5-14　合并后的起动图

3）顺序功能图中的初始步对应于系统等待起动的初始状态，初始步是必不可少的。

4）顺序功能图中一般应有步和有向连线组成的闭环。

5.1.2　梯形图编程的原则

尽管梯形图与继电器电路图在结构形式、元件符号及逻辑控制功能等方面相类似，但它们又有许多不同之处，梯形图有自己的编程规则，大致如下：

① 每一逻辑行总是起于左母线，然后是触点的连接，最后终止于线圈或右母线（右母

线可以不画出)。这仅仅是一般原则。S7 - 200 PLC 的左母线与线圈之间一定要有触点，而线圈与右母线之间则不能有任何触点，如图 5-15 所示。但西门子 S7 - 300 与左母线相连的不一定是触点，而且其线圈不一定与右母线相连。

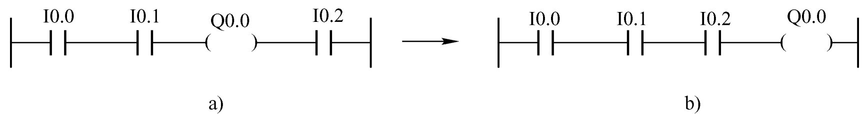


图 5-15 梯形图

② 无论选用哪种机型的 PLC，所用元件的编号必须在该机型的有效范围内。例如 S7 - 200 系列的 PLC 的辅助继电器默认状态下没有 M100.0，若使用就会出错，而 S7 - 300 则有 M100.0。

③ 梯形图中的触点可以任意串联或并联，但继电器线圈只能并联而不能串联。

④ 触点的使用次数不受限制，例如，辅助继电器触点 M0.0 可以在梯形图中出现无限制的次数，但实物继电器的触点一般少于 8 对，只能用有限次。

⑤ 在梯形图中同一线圈只能出现一次。如果在程序中，同一线圈使用了两次或多次，称为“双线圈输出”。对于“双线圈输出”，有些 PLC 将其视为语法错误，绝对不允许；有些 PLC 则将前面的输出视为无效，只有最后一次输出有效（如西门子 PLC）；而有些 PLC 在含有跳转指令或步进指令的梯形图中允许双线圈输出。

⑥ 对于不可编程梯形图必须经过等效变换，变成可编程梯形图，如图 5-16 所示。

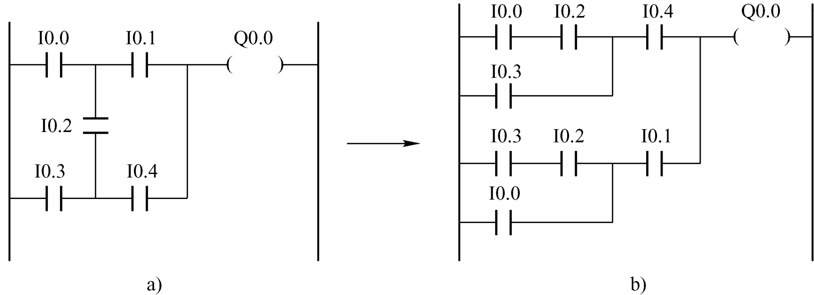


图 5-16 梯形图

⑦ 有几个串联电路相并联时，应将串联触点多的回路放在上方，归纳为“多上少下”的原则，如图 5-17 所示。在有几个并联电路相串联时，应将并联触点多的回路放在左方，归纳为“多左少右”原则，如图 5-18 所示。这样所编制的程序简洁明了，语句较少。但要注意图 5-17a 和图 5-18a 的梯形图逻辑上是正确的。

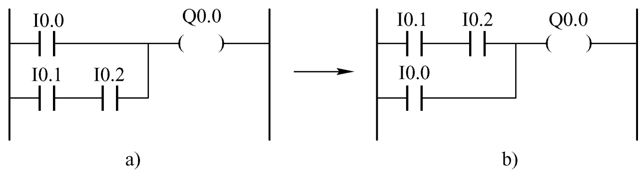


图 5-17 梯形图

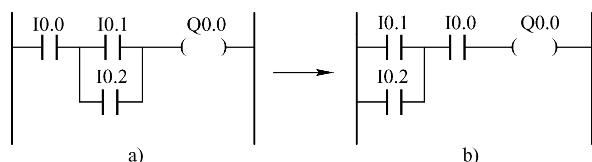


图 5-18　梯形图

⑧ PLC 的输入端所连的电器元件通常使用常开触点，即使与 PLC 对应的继电器－接触器系统原来使用的是常闭触点，改为 PLC 控制时也应转换为常开触点。如图 5-19 所示为继电器－接触器系统控制的电动机的起/停控制，图 5-20 所示为电动机的起/停控制的梯形图，图 5-21 所示为电动机起/停控制的接线图。可以看出：继电器－接触器系统原来使用常闭触点 SB1 和 FR，改用 PLC 控制时，则在 PLC 的输入端变成了常开触点。

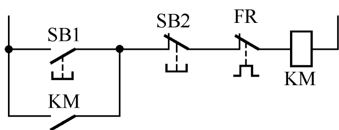


图 5-19　电动机起/停控制图

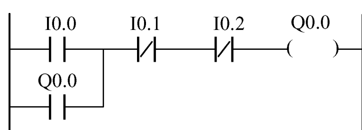


图 5-20　电动机起/停控制的梯形图

【关键点】 图 5-21 的梯形图中 I0.1 和 I0.2 用常闭触点，否则控制逻辑不正确。停止按钮应为常闭触点输入，若读者一定要让 PLC 的输入端的按钮为常闭触点输入也可以（一般不推荐这样使用），但梯形图中 I0.1 和 I0.2 要用常开触点，对于急停按钮必须使用常闭触点，若一定要使用常开触点，从逻辑上讲是可行的，但在某些情况下，有可能因急停按钮不起作用而造成事故，这是读者要特别注意的。另外，一般不推荐将热继电器的常开触点接在 PLC 的输入端，因为这样做占用了宝贵的输入点，最好将热继电器的常闭触点接在 PLC 的输出端，与 KM 线圈串联。

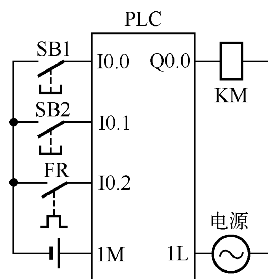


图 5-21　电动机的起/停控制的接线图

5.2　逻辑控制的梯形图编程方法

相同的硬件系统，由不同的人设计，可能设计出不同的程序，有的人设计的程序简洁、可靠，而有的人设计的程序虽然能完成任务，但较复杂，PLC 程序设计是有规律可循的，下面将介绍两种方法：经验设计法和功能图设计法。

1. 经验设计法

就是在一些典型的梯形图的基础上，根据具体对象对控制系统的具体要求，对原有的梯形图进行修改和完善。这种方法适合有一定工作经验的人，这些人手头有现成的资料，特别在产品更新换代时，使用这种方法比较节省时间。下面举例说明这种方法的思路。

【例 5-1】 图 5-22 为小车运输系统的示意图和 I/O 接线图，SQ1、SQ2、SQ3 和 SQ4 是限位开关，小车在 SQ1 处装料，10 s 后右行，到 SQ2 后停止，卸料 10 s 后左行，碰到 SQ1 后

停下装料，就这样不断循环工作，限位开关 SQ3 和 SQ4 的作用是当 SQ2 或者 SQ1 失效时，SQ3 和 SQ4 起保护作用，SB1 是向左行驶起动按钮，SB2 是向右行驶起动按钮，SB3 是停止按钮。

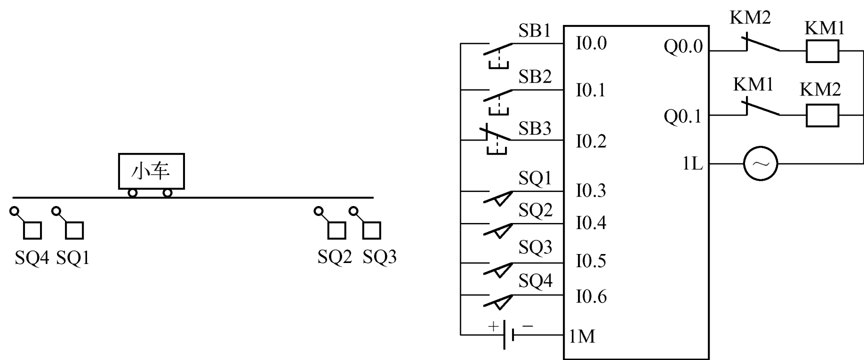


图 5-22　小车运输系统的示意图和 I/O 接线图

【解】

小车左行和右行是不能同时进行的，因此有联锁关系，与电动机的正、反转的梯形图类似，因此先画出电动机正、反转控制的梯形图，如图 5-23 所示，再在这个梯形图的基础上进行修改，增加四个限位开关的输入，增加两个定时器，就变成了如图 5-24 所示的梯形图。

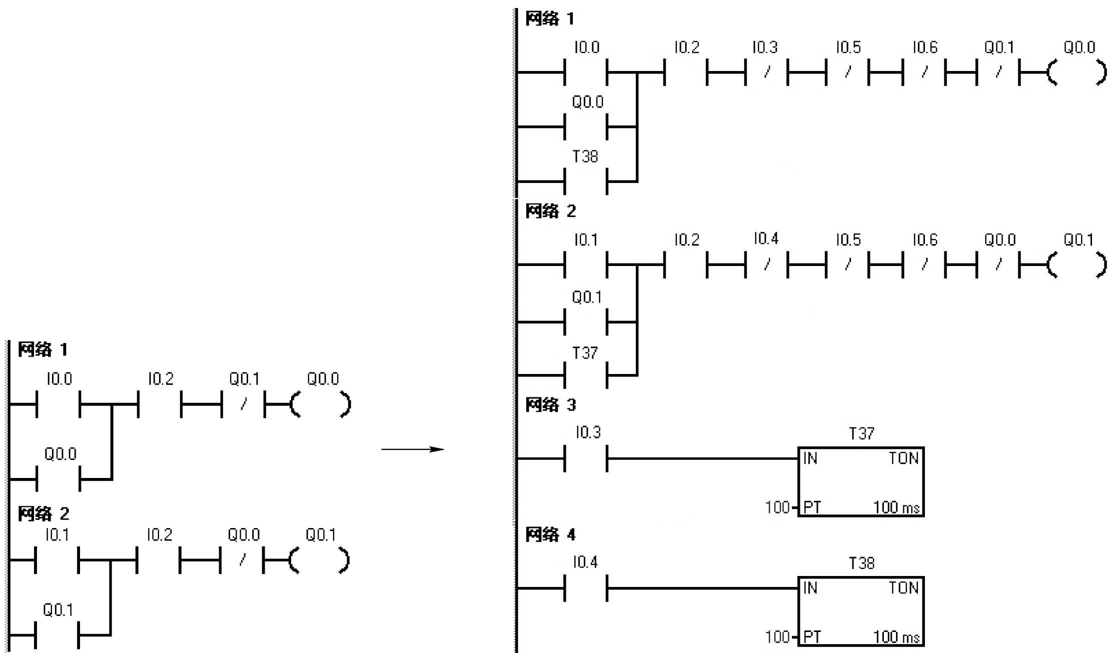


图 5-23　电动机正、反转控制的梯形图

图 5-24　小车运输系统的梯形图

2. 功能图设计法

也称为“起保停”设计法。对于比较复杂的逻辑控制，用经验设计法就不合适，适合

用功能图设计法。功能图设计法无疑是应用最为广泛的设计方法。功能图就是顺序功能图，功能图设计法就是先根据系统的控制要求画出功能图，再根据功能图画梯形图，梯形图可以是基本指令梯形图，也可以是顺控指令梯形图和功能指令梯形图。因此，设计功能图是整个设计过程的关键，也是难点。

（1）起保停设计方法的基本步骤

① 绘制出顺序功能图。要使用“起保停”设计方法设计梯形图时，先要根据控制要求绘制出顺序功能图，其中顺序功能图的绘制在前面章节中已经详细讲解，在此不再重复。

② 写出储存器位的布尔代数式。对应于顺序功能图中的每一个储存器位都可以写出如图 5-25 所示的布尔代数式。图中等号左边的 M_i 为第 i 个储存器位的状态，等号右边的 M_i 为第 i 个储存器位的常开触点，X_i 为第 i 个工步所对应的转换条件，M_{i-1} 为第 i − 1 个储存器位的常开触点，M_{i+1} 为第 i + 1 个储存器位的常闭触点。

③ 写出执行元件的逻辑函数式。执行元件为顺序功能图中的储存器位所对应的动作。一个步通常对应一个动作，输出和对应步的储存器位的线圈并联或者在输出线圈前串联一个对应步的储存器位的常开触点。当功能图中有多个步对应同一动作时，其输出可用这几个步对应的储存器位的“或”来表示，如图 5-26 所示。

$$M_i = (X_i \bullet M_{i-1} + M_i) \bullet \overline{M_{i+1}}$$

图 5-25　存储器位的布尔代数式

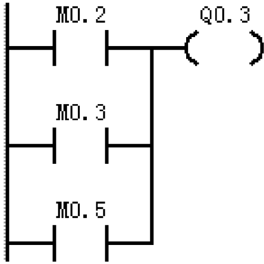


图 5-26　多个步对应同一动作时的梯形图

④ 设计梯形图。在完成前 3 步的基础上，可以顺利设计出梯形图。

（2）利用基本指令编写梯形图程序

用基本指令编写梯形图程序，是最容易被想到的方法，不需要了解较多的指令。采用这种方法编写程序的过程是，先根据控制要求设计正确的功能图，再根据功能图写出正确的布尔表达式，最后根据布尔表达式设计基本指令梯形图。以下用一个例子讲解利用基本指令编写梯形图指令的方法。

【例 5-2】用一个 PLC 控制 4 盏灯，其接线图如图 5-27 所示，4 盏灯的亮灭逻辑描述如下：

① 初始状态时所有的灯都不亮，此时按下 SB1 按钮，灯 HL1 亮；接着按下 SB2 按钮，灯 HL2 亮，HL1 灭，按下 SB3 按钮，灯 HL3 亮，HL2 灭；2 s 后，灯 HL3 仍然亮，灯 HL4 以 1Hz 的频率闪烁，4 s 后，灯 HL3 和 HL4 熄灭，灯 HL1 亮，并如此循环。

② 任何时候，按下 SB4 按钮，所有灯熄灭，并回到初始状态。

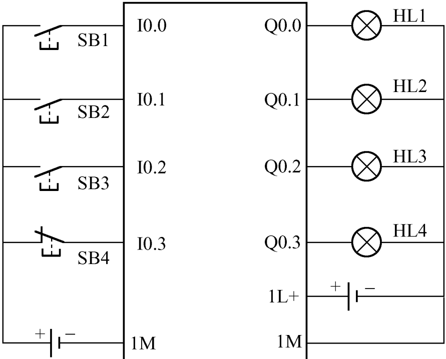


图 5-27　接线图

请画出功能图和梯形图。

【解】

这个逻辑看起来比较复杂，如果不掌握规律，则很难设计出正确的梯形图，一般先根据题意画出功能图，再根据功能图写出布尔表达式，如图 5-28 所示。布尔表达式是有规律的，当前步的步名对应的寄存器（如 M0.1）等于上一步的步名对应的寄存器（M0.0）与上一步的转换条件（I0.1）的乘积，再加上当前步的步名对应的寄存器（M0.1）与下一步的步名对应的寄存器非（$\overline{\text{M0.2}}$）的乘积，其他的布尔表达式的写法类似，最后根据布尔表达式画出梯形图，如图 5-29 所示。在整个过程中，功能图是关键，也是难点，而根据功能图写出布尔表达式和画出梯形图则比较简单。

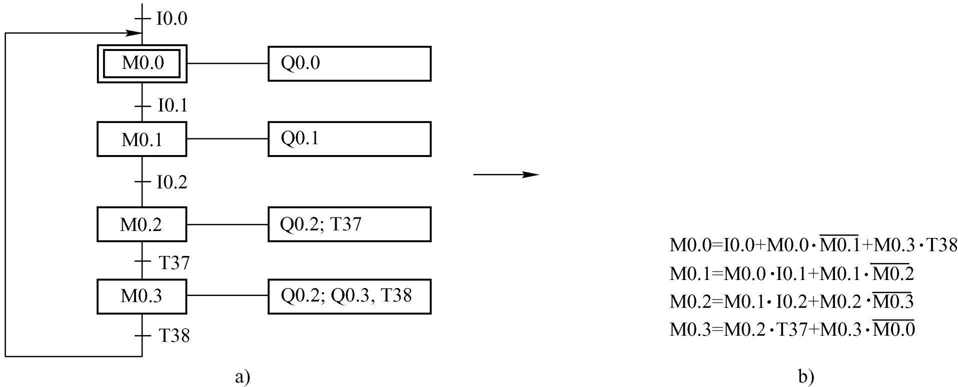


图 5-28　功能图和布尔表达式对应关系图

a）功能图　b）布尔表达式

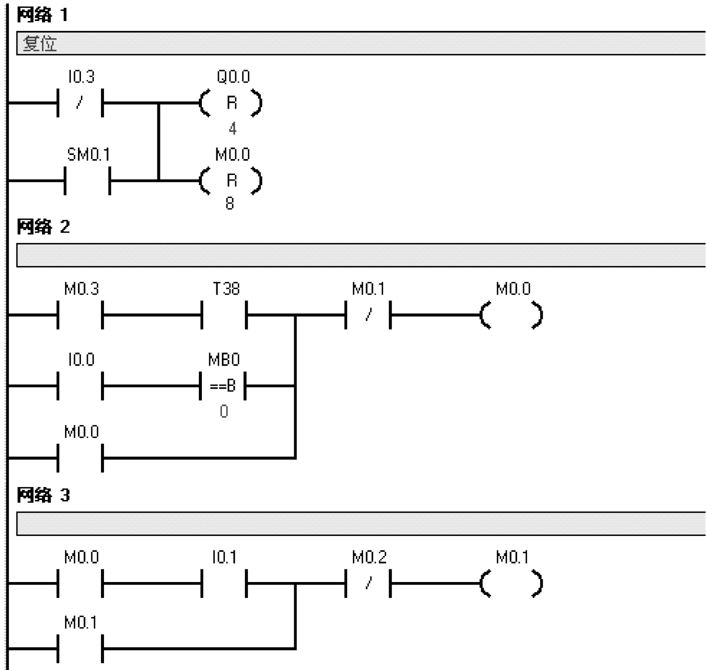


图 5-29　梯形图

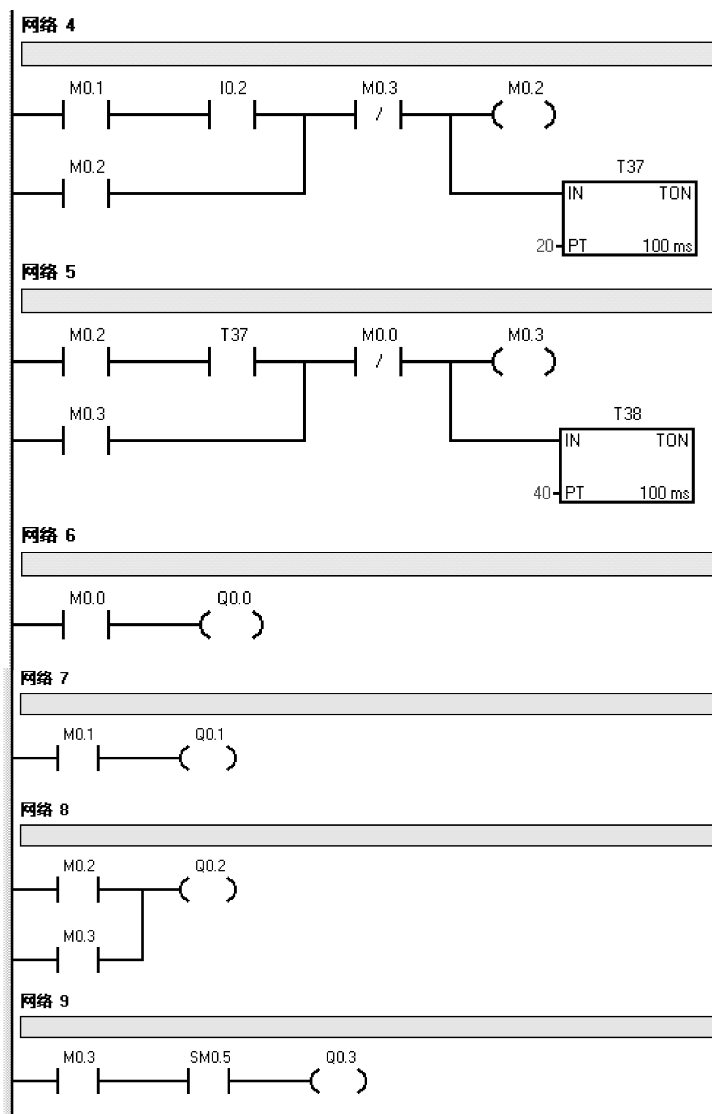


图 5-29　梯形图（续）

【关键点】 这个问题的解决方案中 SB4 是复位按钮，同时也起停止按钮的作用，因此，接线图上最好使用常闭触点。

图 5-29 梯形图的网络 2 中有一个 MB0 = 0 导通的条件是确保在非起始步，SB1 按钮不起作用，也就是说除第一步外，在其他几步中按 SB1 按钮，都不会起作用。

【例 5-3】 步进电动机是一种将电脉冲信号转换为电动机旋转角度的执行机构。当步进驱动器接收到一个脉冲，就驱动步进电动机按照设定的方向旋转一个固定的角度（称为步距角）。因此步进电动机是按照固定的角度一步一步转动的。因此可以通过脉冲数量控制步进电动机的运行角度，并通过相应的装置，控制运动的过程。对于五相十拍步进电动机，其控制要求为：按下起停按钮，定子磁极 A 通电，1 s 后 A、B 同时通电；再过 1 s，B 通电，同时 A 失电；再过 1 s，B、C 同时通电……，以此类推，其通电过程如图 5-30 所示。

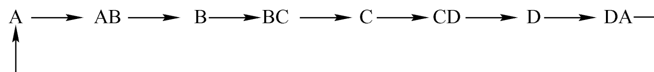


图 5-30　通电过程图

【解】

根据题意很容易画出功能图，如图 5-31 所示。根据功能图编写梯形图程序如图 5-32 所示。

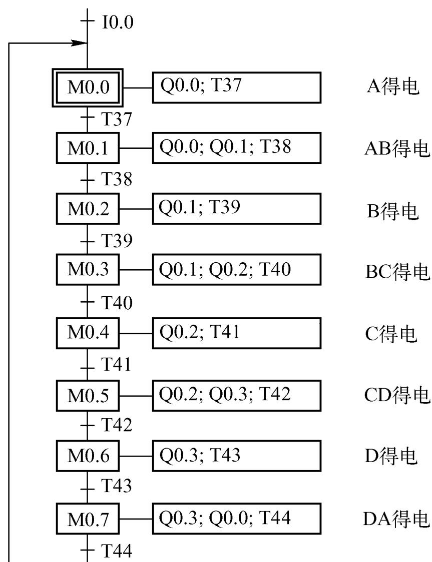


图 5-31　功能图

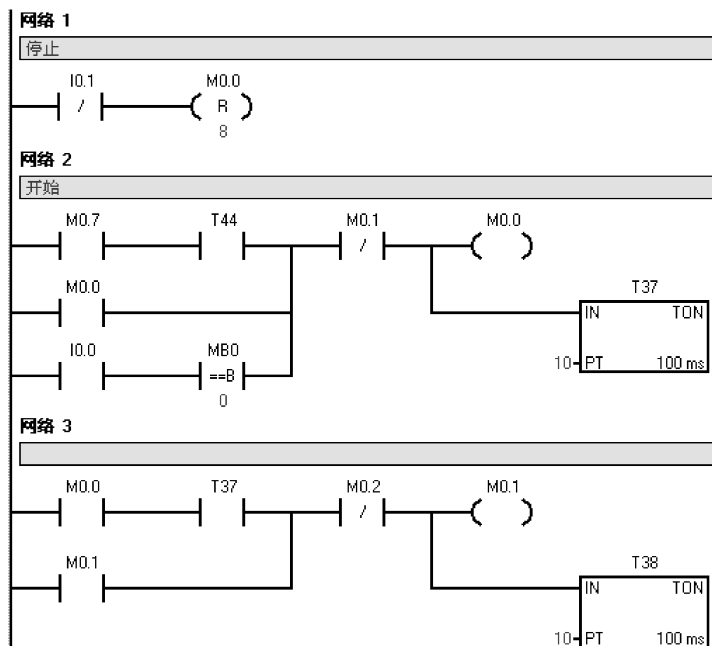


图 5-32　梯形图

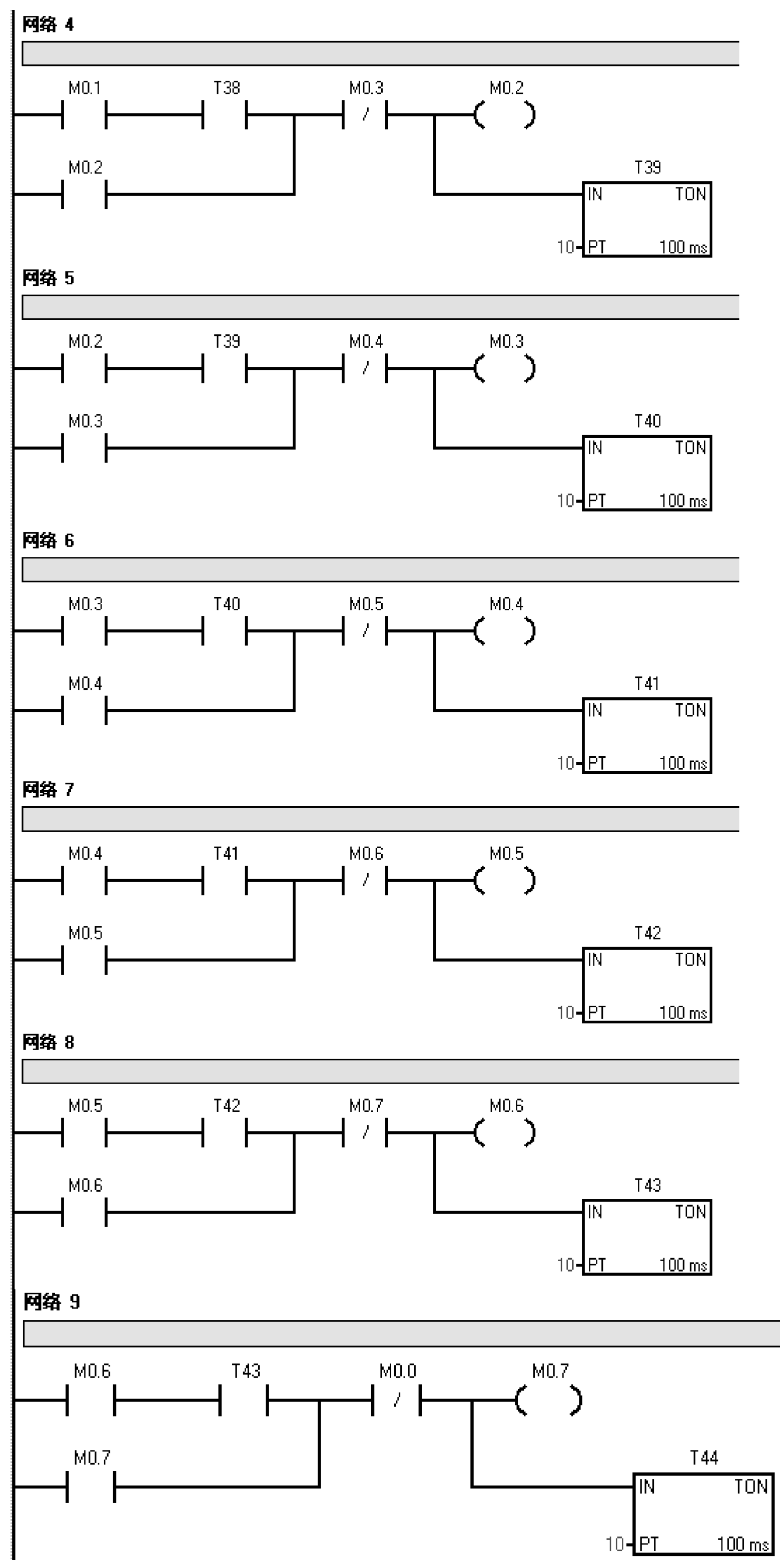

网络 4
M0.1
T38
M0.3
M0.2
M0.2
T39
IN
TON
10
PT
100 ms
网络 5
M0.2
T39
M0.4
M0.3
M0.3
T40
IN
TON
10
PT
100 ms
网络 6
M0.3
T40
M0.5
M0.4
M0.4
T41
IN
TON
10
PT
100 ms
网络 7
M0.4
T41
M0.6
M0.5
M0.5
T42
IN
TON
10
PT
100 ms
网络 8
M0.5
T42
M0.7
M0.6
M0.6
T43
IN
TON
10
PT
100 ms
网络 9
M0.6
T43
M0.0
M0.7
M0.7
T44
IN
TON
10
PT
100 ms

图 5-32　梯形图（续）

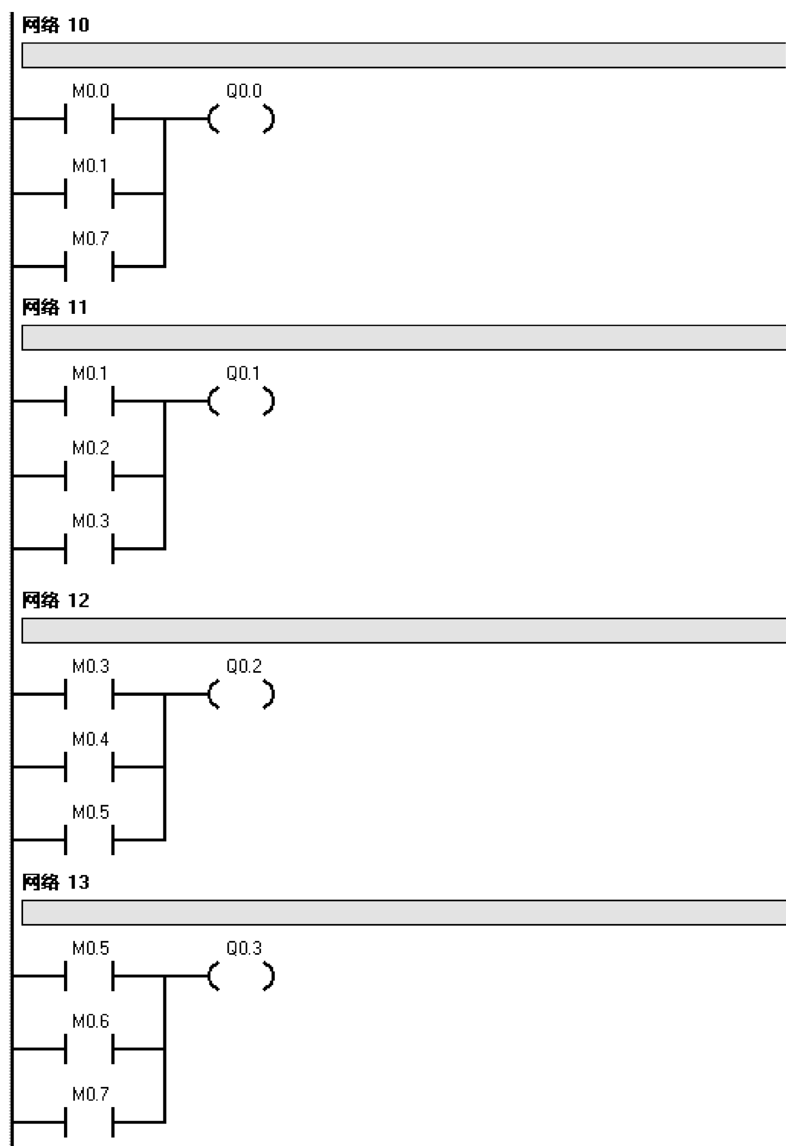


图 5-32　梯形图（续）

（3）利用顺控指令编写逻辑控制程序

功能图和顺控指令梯形图有一一对应关系，利用顺控指令编写逻辑控制程序有固定的模式，顺控指令是专门为逻辑控制设计的指令，利用顺控指令编写逻辑控制程序是非常合适的。以下用一个例子讲解利用顺控指令编写逻辑控制程序。

【例 5-4】 用顺控指令编写例 5-2 的程序。

【解】

功能图如图 5-33 所示，程序如图 5-34 所示。

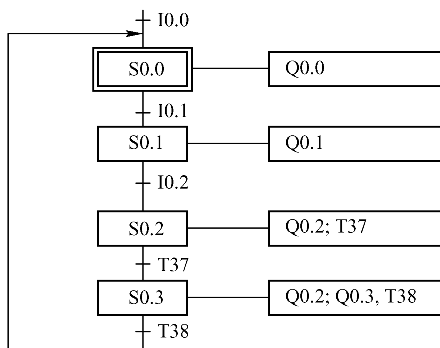


图 5-33　功能图

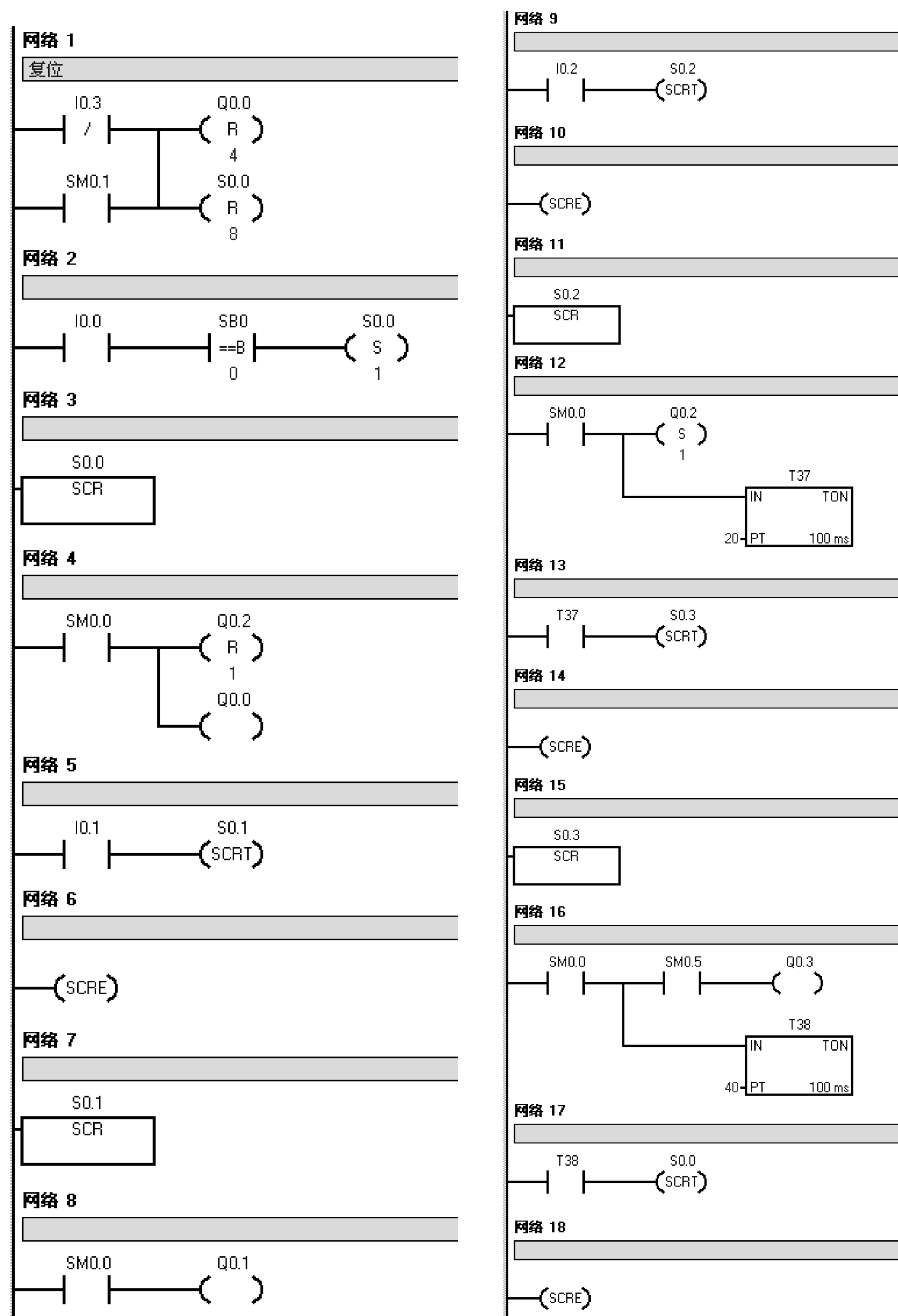


图 5-34 程序

【例 5-5】用顺控指令编写例 5-3 的程序。

【解】

功能图如图 5-35 所示，程序如图 5-36 所示。

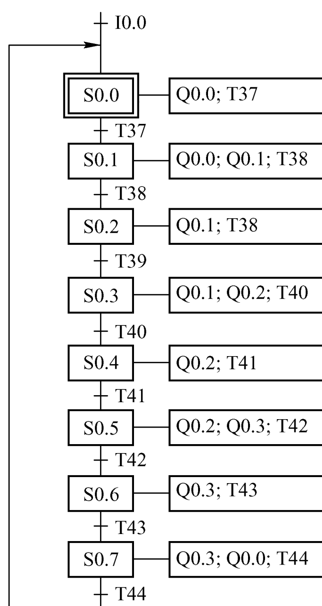


图 5-35　功能图

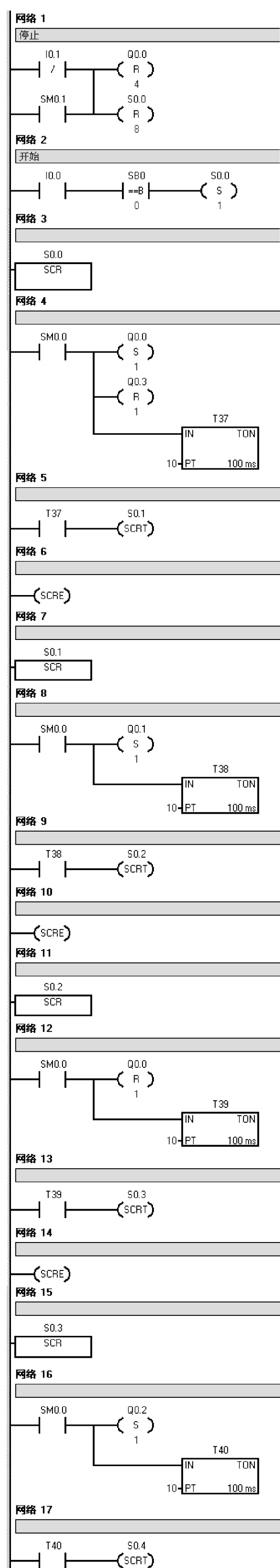


图 5-36　梯形图

（4）利用功能指令编写逻辑控制程序

西门子的功能指令有许多特殊的功能，其中功能指令中的移位指令和循环指令非常适合用于顺序控制，用这些指令编写程序简洁而且可读性强。以下用一个例子讲解利用功能指令编写逻辑控制程序。

【例 5-6】用功能指令编写例 5-2 的程序。

【解】

梯形图如图 5-37 所示。

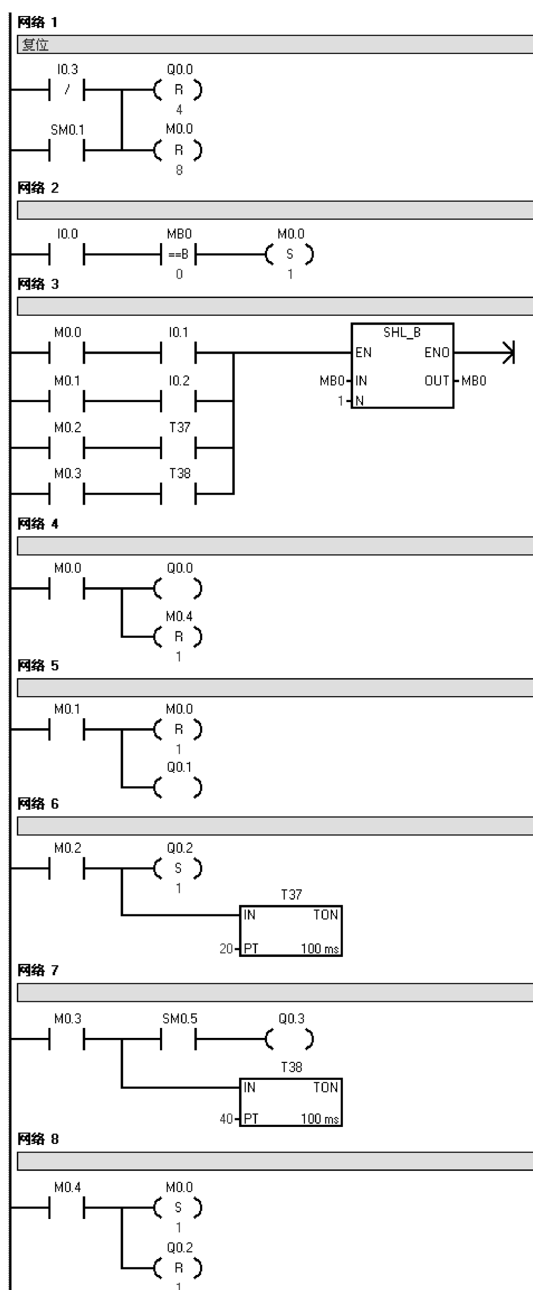


图 5-37　梯形图

【例 5-7】用功能指令编写例 5-3 的程序。

【解】

梯形图如图 5-38 所示。

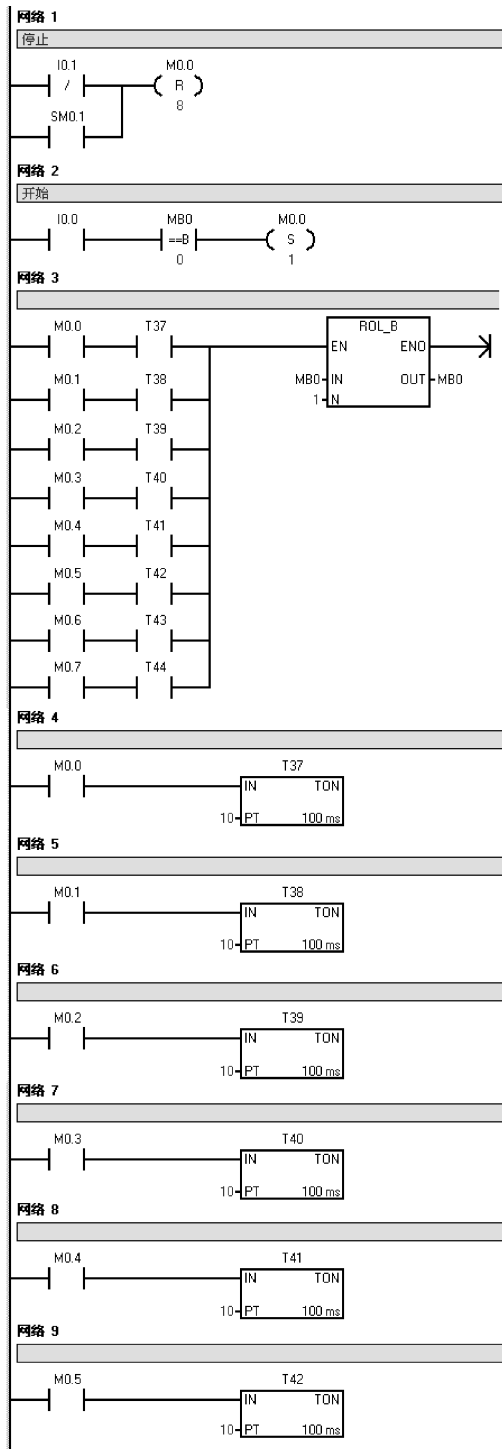


图 5-38　梯形图

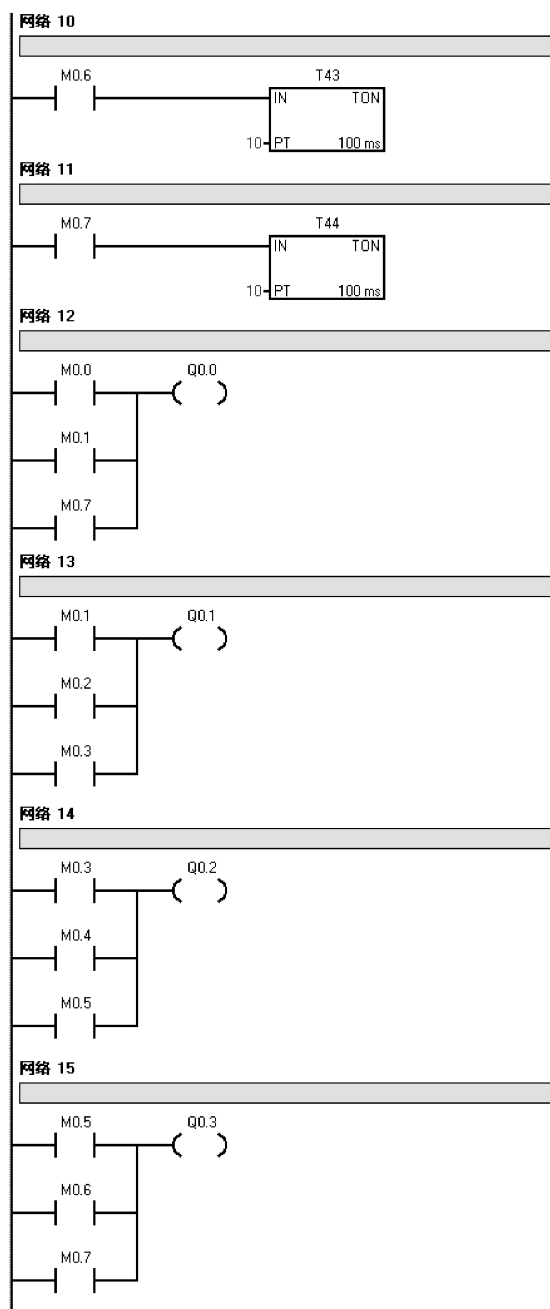


图 5-38　梯形图（续）

（5）利用复位/置位指令编写逻辑控制程序

复位/置位指令是常用指令，用复位/置位指令编写程序简洁而且可读性强。以下用一个例子讲解利用复位/置位指令编写逻辑控制程序。

【例 5-8】用复位和置位指令编写例 5-2 的程序。

【解】

梯形图如图 5-39 所示。

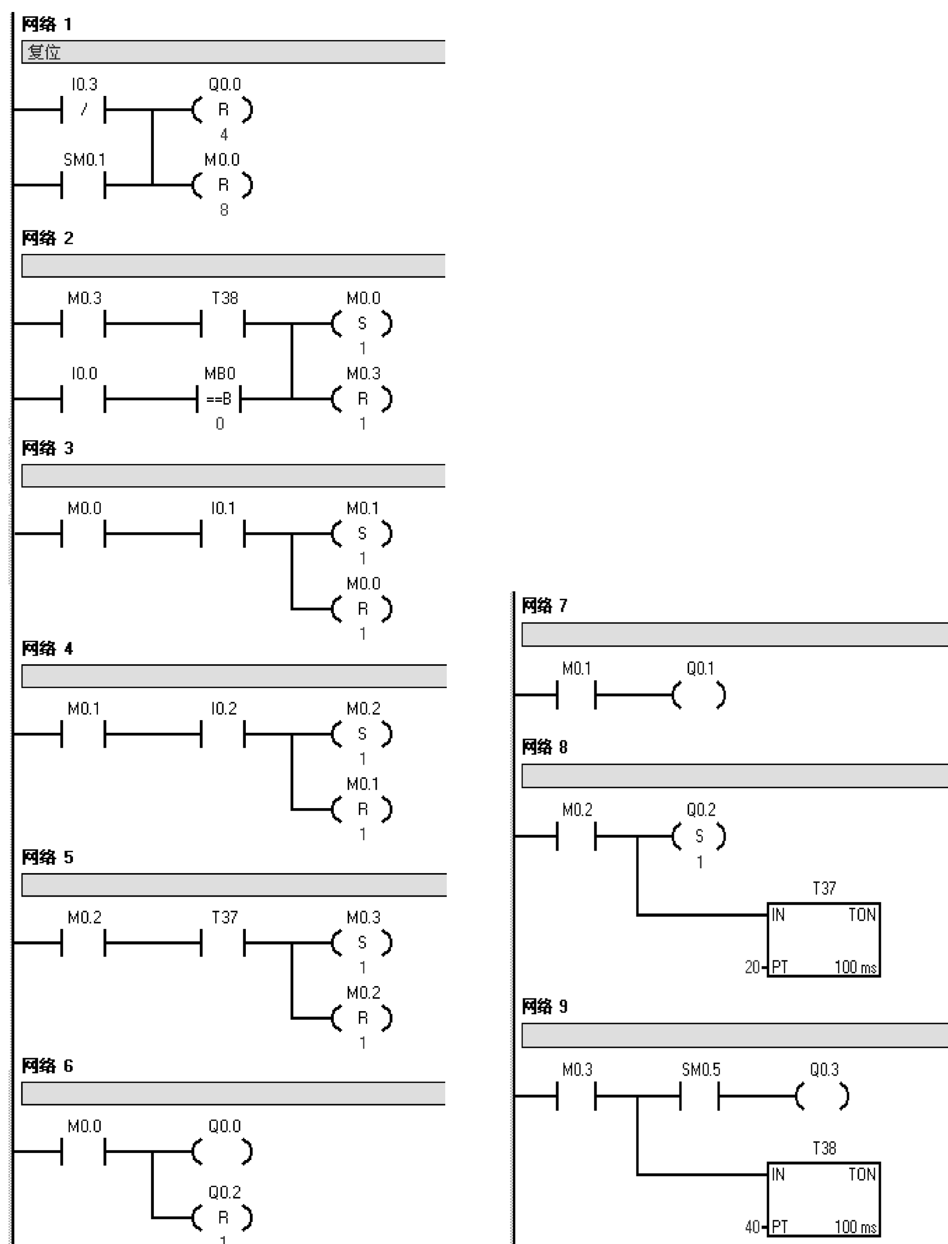


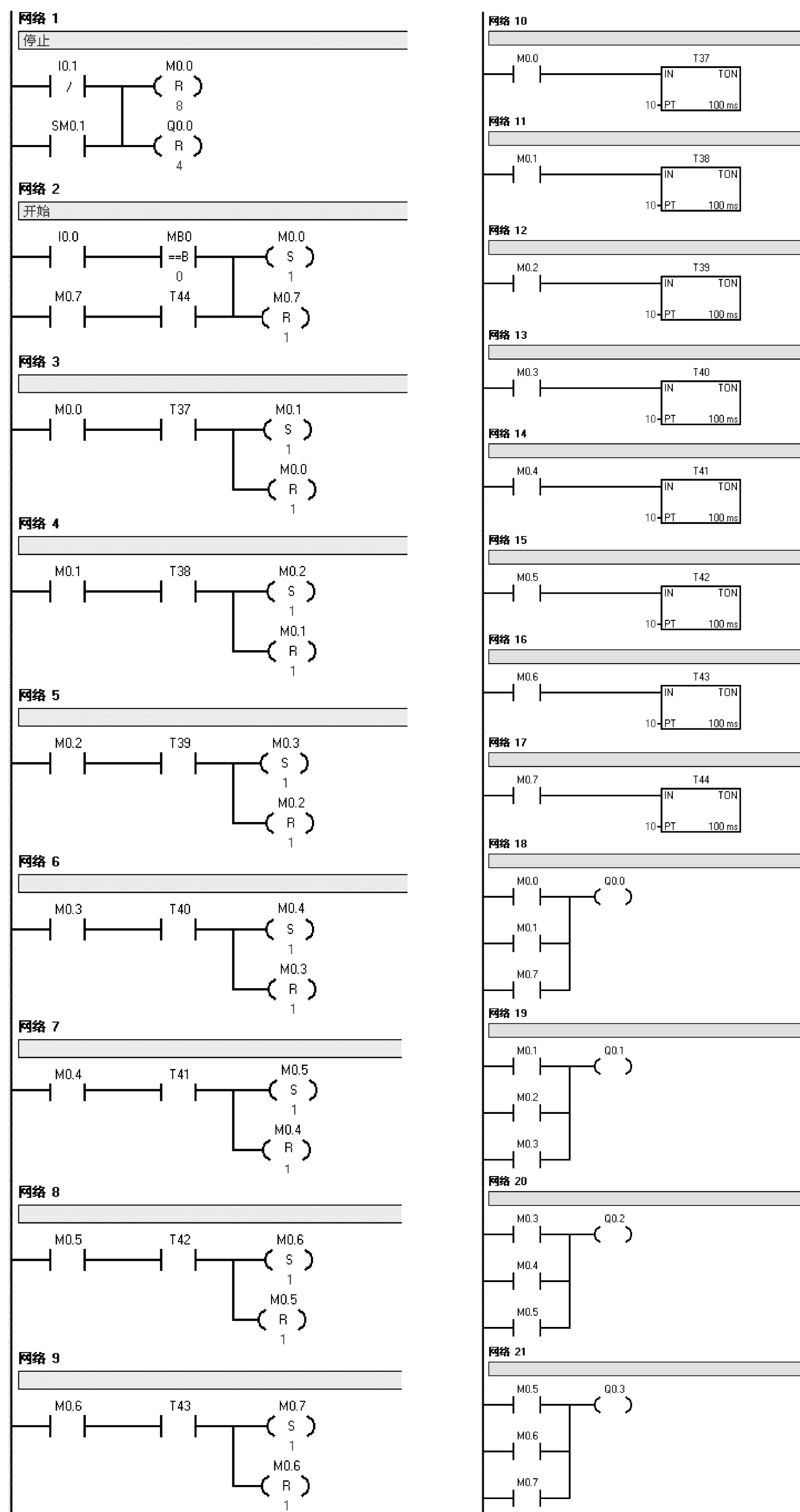
图 5-39　程序

【例 5-9】 用置位/复位指令编写例 5-3 的程序。

【解】

梯形图如图 5-40 所示。

至此，同一个顺序控制的问题使用了基本指令、顺控指令（有的 PLC 称为步进梯形图指令）、复位/置位指令和功能指令四种解决方案编写程序。四种解决方案的编程都有各自的几乎固定的步骤，但有一步是相同的，那就是首先都要画功能图。四种解决方案没有好坏之分，读者可以根据自己的喜好选用。在下一节的“项目实施”中，读者可以模仿以上四种方法中的一种来编写程序。



网络 1
停止
I0.1
M0.0
R
8
SM0.1
Q0.0
R
4
网络 2
开始
I0.0
MB0
==B
0
M0.0
S
1
M0.7
T44
M0.7
R
1
网络 3
M0.0
T37
M0.1
S
1
M0.0
R
1
网络 4
M0.1
T38
M0.2
S
1
M0.1
R
1
网络 5
M0.2
T39
M0.3
S
1
M0.2
R
1
网络 6
M0.3
T40
M0.4
S
1
M0.3
R
1
网络 7
M0.4
T41
M0.5
S
1
M0.4
R
1
网络 8
M0.5
T42
M0.6
S
1
M0.5
R
1
网络 9
M0.6
T43
M0.7
S
1
M0.6
R
1
网络 10
M0.0
T37
IN TON
10 PT 100 ms
网络 11
M0.1
T38
IN TON
10 PT 100 ms
网络 12
M0.2
T39
IN TON
10 PT 100 ms
网络 13
M0.3
T40
IN TON
10 PT 100 ms
网络 14
M0.4
T41
IN TON
10 PT 100 ms
网络 15
M0.5
T42
IN TON
10 PT 100 ms
网络 16
M0.6
T43
IN TON
10 PT 100 ms
网络 17
M0.7
T44
IN TON
10 PT 100 ms
网络 18
M0.0
Q0.0
M0.1
M0.7
网络 19
M0.1
Q0.1
M0.2
M0.3
网络 20
M0.3
Q0.2
M0.4
M0.5
网络 21
M0.5
Q0.3
M0.6
M0.7

图 5–40 梯形图

以下再举 2 个例子说明逻辑控制的编程方法。

【例 5-10】 液体混合装置示意图如图 5-41 所示，上限位、下限位和中限位液位传感器被液体淹没时为 1 状态，电磁阀 A、B、C 的线圈通电时，阀门打开，电磁阀 A、B、C 的线圈断电时，阀门关闭。在初始状态时容器是空的，各阀门均关闭，各传感器均为 0 状态。按下起动按钮后，打开电磁阀 A，液体 A 流入容器，中限位开关变为 ON 时，关闭 A，打开阀 B，液体 B 流入容器。液面上升到上限位开关，关闭阀门 B，电动机 M 开始运行，搅拌液体，30 s 后停止搅动，打开电磁阀 C，放出混合液体，当液面下降到下限位开关之后，过 3 s，容器放空，关闭电磁阀 C，打开电磁阀 A，又开始下一个周期的操作。按停止按钮，当前工作周期结束后，才能停止工作，按急停按钮可立即停止工作。请绘制功能图，设计梯形图。

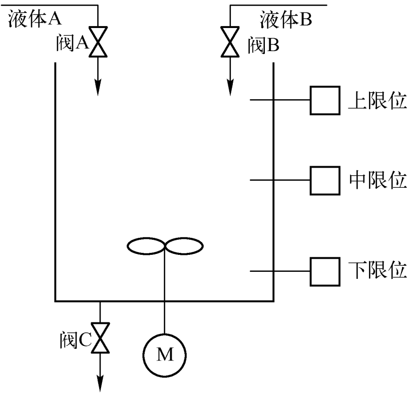


图 5-41　液体混合装置

【解】

液体混合的 PLC 的 I/O 分配见表 5-1。

表 5-1　PLC 的 I/O 分配表

输入			输出		
名　　称	符　　号	输　入　点	名　　称	符　　号	输　出　点
开始按钮	SB1	I0.0	电磁阀 A	YV1	Q0.0
停止按钮	SB2	I0.1	电磁阀 B	YV2	Q0.1
急停	SB3	I0.2	电磁阀 C	YV3	Q0.2
上限位传感器	SQ1	I0.3	电动机	M	Q0.3
中限位传感器	SQ2	I0.4			
下限位传感器	SQ3	I0.5			

电气系统的原理图如图 5-42 所示，功能图如图 5-43 所示，梯形图如图 5-44 所示。

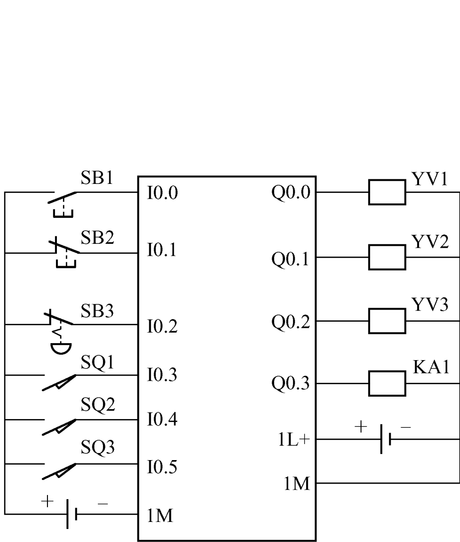


图 5-42　原理图

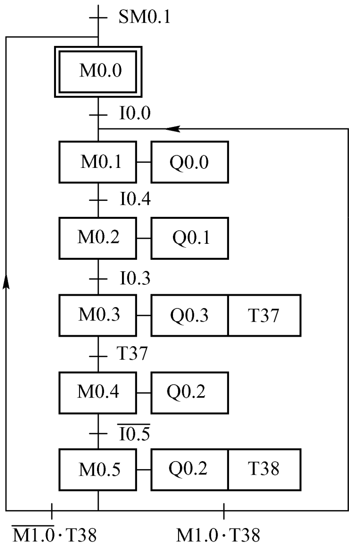


图 5-43　功能图

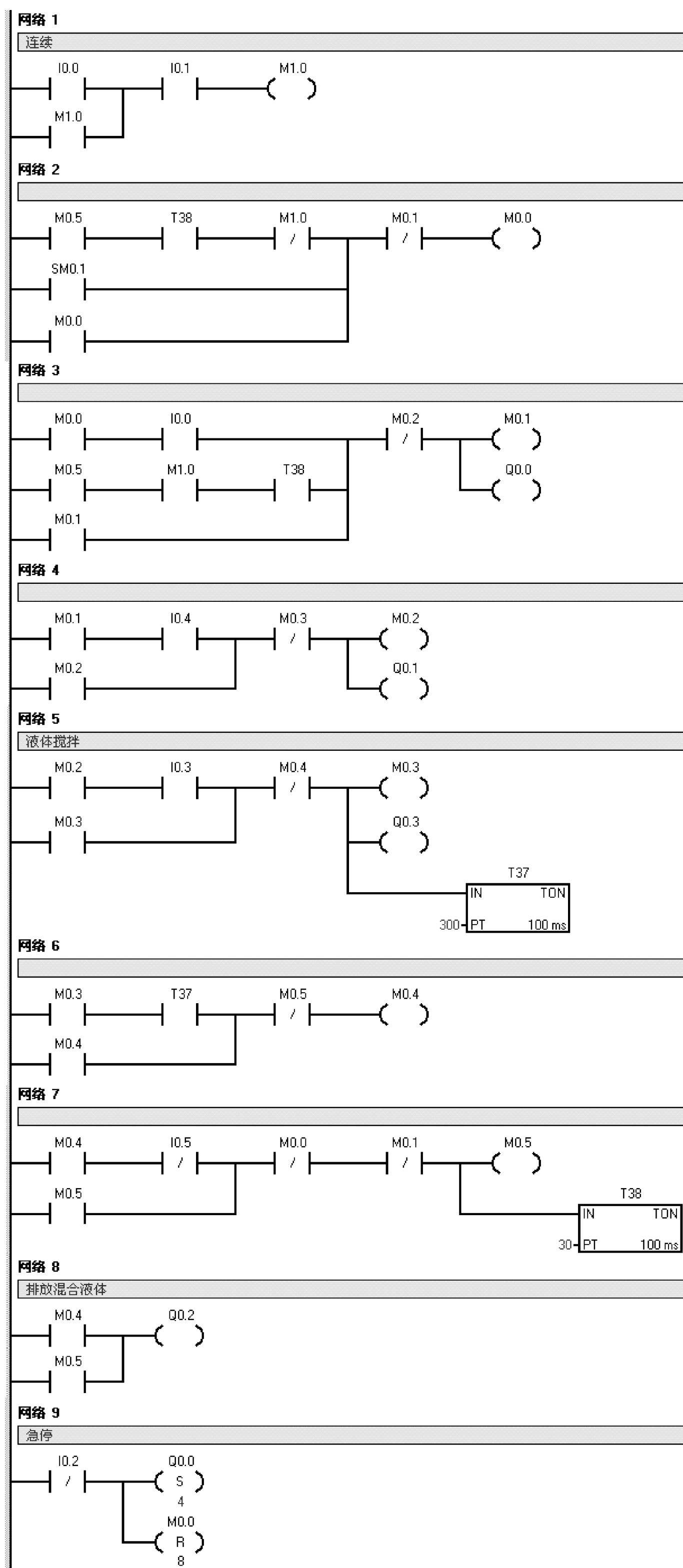

网络 1
连续
I0.0
I0.1
M1.0
M1.0
网络 2
M0.5
T38
M1.0
M0.1
M0.0
SM0.1
M0.0
网络 3
M0.0
I0.0
M0.2
M0.1
M0.5
M1.0
T38
Q0.0
M0.1
网络 4
M0.1
I0.4
M0.3
M0.2
M0.2
Q0.1
网络 5
液体搅拌
M0.2
I0.3
M0.4
M0.3
M0.3
Q0.3
T37
IN
TON
300
PT
100 ms
网络 6
M0.3
T37
M0.5
M0.4
M0.4
网络 7
M0.4
I0.5
M0.0
M0.1
M0.5
M0.5
T38
IN
TON
30
PT
100 ms
网络 8
排放混合液体
M0.4
Q0.2
M0.5
网络 9
急停
I0.2
Q0.0
S
4
M0.0
R
8

图 5-44　梯形图

【例 5-11】某钻床用 2 个钻头同时钻 2 个孔，开始自动运行之前，2 个钻头在最上面，上限位开关 I0.3 和 I0.5 为 ON。操作人员放好工件后，按起动按钮 I0.0。工件被夹紧后，2 个钻头同时开始工作，钻到由限位开关 I0.2 和 I0.4 设定的深度时分别上行，回到由限位开关 I0.3 和 I0.5 设定的起始位置时，分别停止上行。当 2 个钻头都到起始位置后，工件松开，工件松开后，加工结束，系统回到初始状态。钻床的加工示意图如图 5-45 所示，请设计功能图和梯形图。

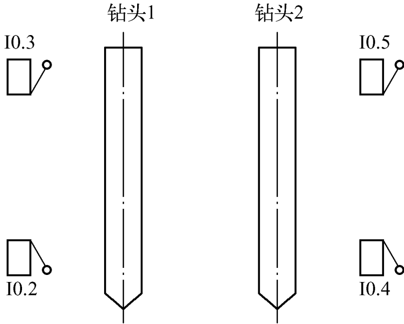


图 5-45　钻床加工示意图

【解】

钻床的 PLC 的 I/O 分配见表 5-2。

表 5-2　PLC 的 I/O 分配表

输　入			输　出		
名　称	符　号	输　入　点	名　称	符　号	输　出　点
开始按钮	SB1	I0.0	夹具夹紧	KA1	Q0.0
停止按钮	SB2	I0.1	钻头 1 下降	KA2	Q0.1
钻头 1 上限位开关	SQ1	I0.2	钻头 1 上升	KA3	Q0.2
钻头 1 下限位开关	SQ2	I0.3	钻头 2 下降	KA4	Q0.3
钻头 2 上限位开关	SQ3	I0.4	钻头 2 上升	KA5	Q0.4
钻头 2 下限位开关	SQ4	I0.5	夹具松开	KA6	Q0.5
夹紧限位开关	SQ5	I0.6			
松开下限位开关	SQ6	I0.7			

电气系统的原理图如图 5-46 所示，功能图如图 5-47 所示，梯形图如图 5-48 所示。

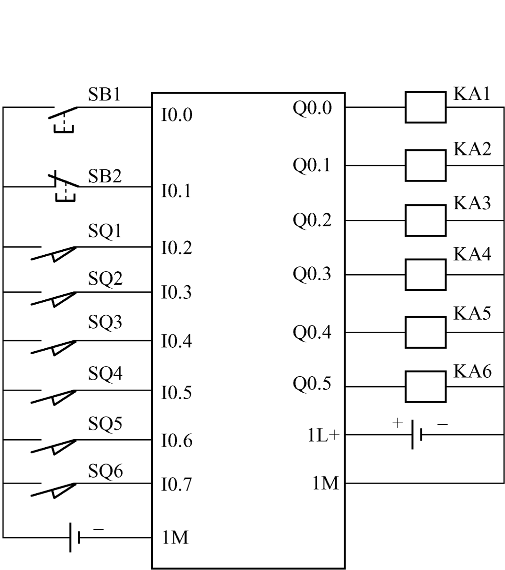


图 5-46　原理图

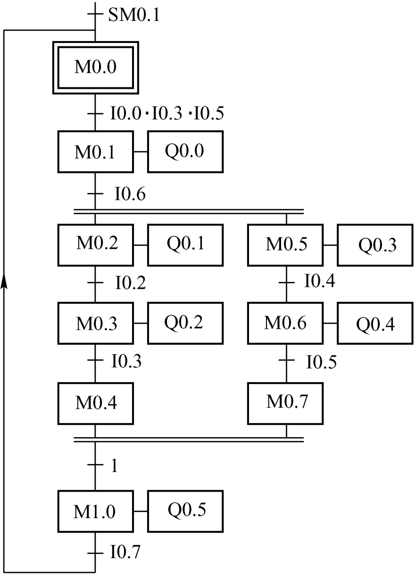


图 5-47　功能图

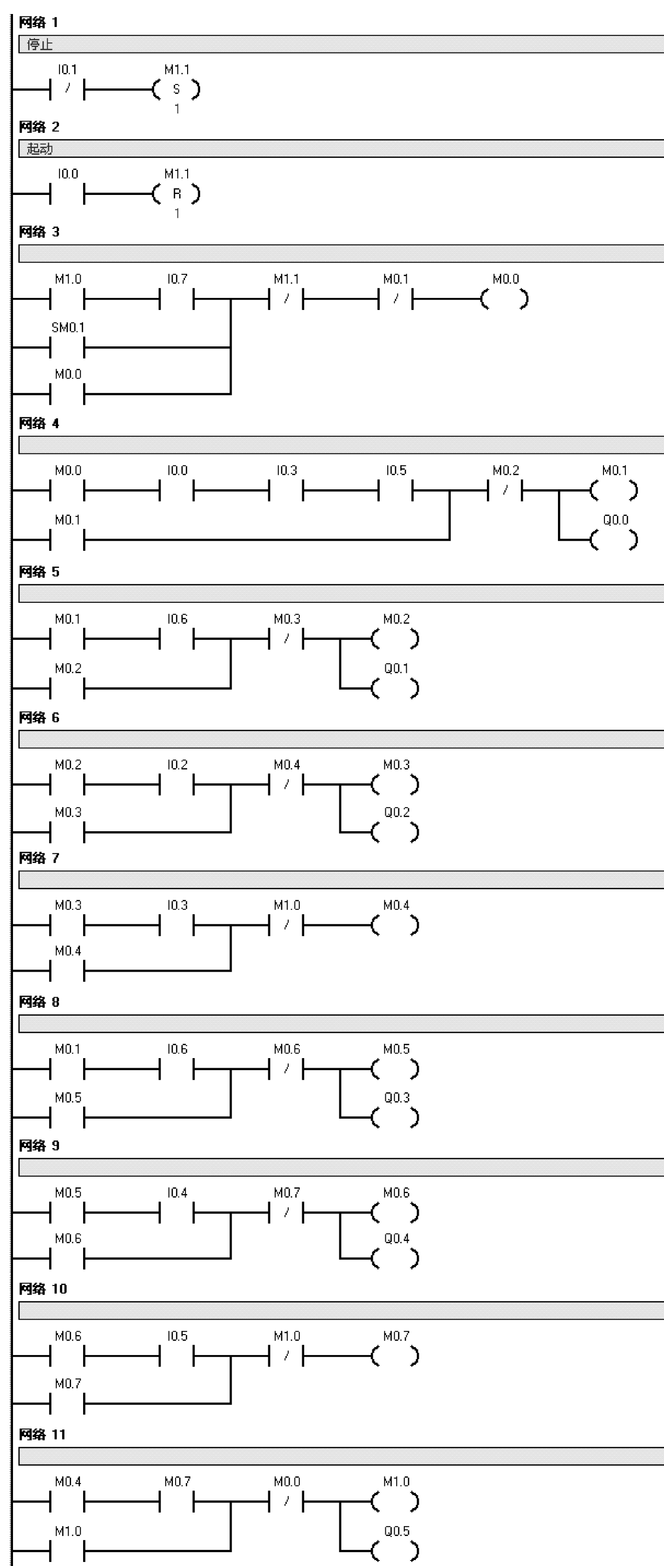

网络 1
停止
I0.1
M1.1
S
1
网络 2
起动
I0.0
M1.1
R
1
网络 3
M1.0
I0.7
M1.1
M0.1
M0.0
SM0.1
M0.0
网络 4
M0.0
I0.0
I0.3
I0.5
M0.2
M0.1
M0.1
Q0.0
网络 5
M0.1
I0.6
M0.3
M0.2
M0.2
Q0.1
网络 6
M0.2
I0.2
M0.4
M0.3
M0.3
Q0.2
网络 7
M0.3
I0.3
M1.0
M0.4
M0.4
网络 8
M0.1
I0.6
M0.6
M0.5
M0.5
Q0.3
网络 9
M0.5
I0.4
M0.7
M0.6
M0.6
Q0.4
网络 10
M0.6
I0.5
M1.0
M0.7
M0.7
网络 11
M0.4
M0.7
M0.0
M1.0
M1.0
Q0.5

图 5-48　梯形图

小　　结

（1）要学会设计功能图，这是非常重要的，因为比较复杂的程序都要根据功能图编写，因此如果不能根据工程的要求设计出功能图，很难编写正确的程序。

（2）程序的编写方法，同一个顺序控制的问题使用了基本指令、复位/置位指令、顺控指令和功能指令四种解决方案编写程序。四种解决方案的编程都有各自几乎固定的步骤，但有一步是相同的，那就是首先都要画功能图。四种解决方案没有好坏之分，读者可以根据自己的喜好选用。相对而言，用置位/复位指令编写程序更加容易被初学者掌握。

（3）在没有硬件的情况下，验证 S7－200 程序正确性最好的办法是使用仿真软件。

习　　题

1. 在功能图中，什么是步、活动步、动作和转换条件？

2. 设计功能图要注意什么？

3. 编写梯形图程序要注意哪些问题？

4. 编写逻辑控制梯形图程序有哪些常用的方法？

5. 用 I0.0 控制 Q0.0、Q0.1 和 Q0.2，要求：I0.0 闭合两次，Q0.0 亮，I0.0 再闭合两次，Q0.1 亮，I0.0 再闭合两次，Q0.2 亮，I0.0 再闭合一次，Q0.0、Q0.1 和 Q0.2 灭，如此循环，请先设计功能图和接线图，再编写程序，要求用基本指令、移位指令和复位/置位指令编写。

6. 要求用移位指令使 8 盏灯以 0.2 s 的速度自左向右亮起，到达最右侧后，再自右向左回到最左侧。如此循环，I0.0 按钮为起动按钮，I0.1 为停止按钮，请先设计功能图和接线图，再编写程序，要求用基本指令、移位指令和复位/置位指令编写。

7. 正反转的程序如图 5-49 所示，请用仿真软件，进行仿真，验证程序是否正确？

8. 根据如图 5-50 所示的功能图编写程序。

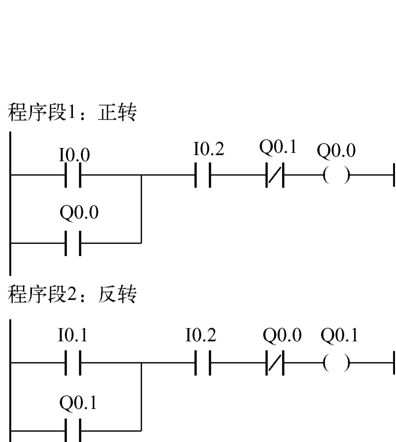


图 5-49　程序

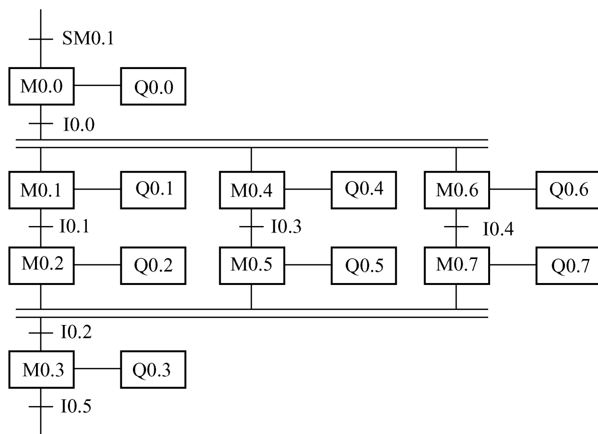


图 5-50　功能图

9. 根据如图 5-51 所示的功能图编写程序。

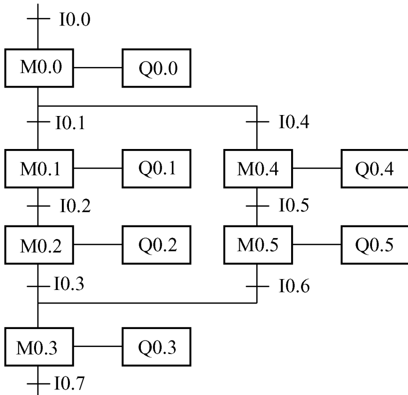


图 5-51 功能图

10. 机械手的工作示意图如图 5-52 所示，当合上按钮 I0. 4，机械手将工件从 A 点搬运到 B 点，然后返回 A 点，再如此循环，任何时候合上按钮 I0. 5 时系统复位回到原点，请编写控制程序。

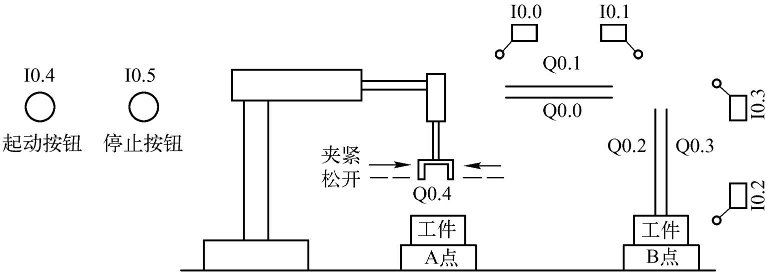


图 5-52 机械手示意图

第 6 章　S7 - 200 PLC 的通信及其应用

本章介绍 S7 - 200 PLC 的通信的基础知识，并用实例介绍 S7 - 200 PLC 的 PPI、MPI、PROFIBUS、自由口、Modbus 和以太网通信。本章的内容较多，而且较难。

6.1　通信基础知识

PLC 的通信包括 PLC 与 PLC 之间的通信、PLC 与上位计算机之间的通信以及和其他智能设备之间的通信。PLC 与 PLC 之间通信的实质就是计算机的通信，使得众多独立的控制任务构成一个控制工程整体，形成模块控制体系。PLC 与计算机连接组成的网络，PLC 用于控制工业现场，计算机用于编程、显示和管理等任务，构成“集中管理、分散控制”的分布式控制系统（DCS）。

6.1.1　通信的基本概念

1. 串行通信与并行通信

串行通信和并行通信是两种不同的数据传输方式。

并行通信就是将一个 8 位数据（或 16 位、32 位）的每一个二进制位采用单独的导线进行传输，并将传送方和接收方进行并行连接，一个数据的各二进制位可以在同一时间一次传送。例如，老式打印机的打印口和计算机的通信就是并行通信。并行通信的特点是一个周期里可以一次传输多位数据，其连线的电缆多，因此长距离传送时成本高。

串行通信就是通过一对导线将发送方与接收方进行连接，传输数据的每个二进制位，按照规定顺序在同一导线上依次发送与接收。例如，优盘的 USB 接口就是串行通信。串行通信的特点是通信控制复杂，通信电缆少，因此与并行通信相比，成本低。串行通信是一种趋势，随着串行通信速率的提高，以往使用并行通信的场合，现在完全或部分被串行通信取代，如打印机的通信，现在基本被串行通信取代，再如个人计算机硬盘的数据通信，现在也已经被串行通信取代。

2. 异步通信与同步通信

异步通信与同步通信也称为异步传送与同步传送，这是串行通信的两种基本信息传送方式。从用户的角度上说，两者最主要的区别在于通信方式的“帧”不同。

异步通信方式又称起止方式。它在发送字符时，要先发送起始位，然后是字符本身，最后是停止位，字符之后还可以加入奇偶校验位。异步通信方式具有硬件简单、成本低的特点，主要用于传输速率低于 19.2 kbit/s 以下的数据通信。

同步通信方式在传递数据的同时，也传输时钟同步信号，并始终按照给定的时刻采集数据。其传输数据的效率高，硬件复杂，成本高，一般用于传输速率高于 20 kbit/s 以上的数据通信。

3. 单工、双工与半双工

单工、双工与半双工是通信中描述数据传送方向的专用术语。分别介绍如下：

① 单工（Simplex）：指数据只能实现单向传送的通信方式，一般用于数据的输出，不可以进行数据交换。

② 全双工（Full Simplex）：也称双工，指数据可以进行双向数据传送，同一时刻既能发送数据，也能接收数据。通常需要两对双绞线连接，通信线路成本高。例如，RS－422 和 RS－232C 就是“全双工”通信方式。

③ 半双工（Half Simplex）：指数据可以进行双向数据传送，同一时刻，只能发送数据或者接收数据。通常需要一对双绞线连接，与全双工相比，通信线路成本低。例如，RS－485 只用一对双绞线时就是“半双工”通信方式。

6.1.2 RS－485 标准串行接口

1. RS－485 接口

RS－485 接口是在 RS－422 基础上发展起来的一种 EIA 标准串行接口，采用“平衡差分驱动”方式。RS－485 接口满足 RS－422 的全部技术规范，可用于 RS－422 通信。RS－485 接口通常采用 9 针连接器。RS－485 接口的引脚功能参见表 6-1。

表 6-1 RS－485 接口的引脚功能

PLC 侧引脚	信号代号	信号功能
1	SG 或 GND	机壳接地
2	+24 V 返回	逻辑地
3	RXD＋或 TXD＋	RS－485 的 B，数据发送/接收＋端
4	请求－发送	RTS(TTL)
5	+5 V 返回	逻辑地
6	+5 V	+5 V，100 Ω 串联电阻
7	+24 V	+24 V
8	RXD－或 TXD－	RS－485 的 A，数据发送/接收－端
9	不适用	10 位协议选择（输入）

2. 西门子的 PLC 连线

西门子 PLC 的 PPI 通信、MPI 通信和 PROFIBUS－DP 现场总线通信的物理层都是 RS－485 通信，而且采用都是相同的通信线缆和专用网络接头。西门子提供两种网络接头，即标准网络接头和包括编程端口接头，可方便地将多台设备与网络连接，编程端口允许用户将编程站或 HMI 设备与网络连接，而不会干扰任何现有网络连接。编程端口接头通过编程端口传送所有来自 S7－200 CPU 的信号（包括电源针脚），这对于连接由 S7－200 CPU（例如 SIMATI C 文本显示）供电的设备尤其有用。标准网络接头的编程端口接头均有两套终端螺钉，用于连接输入和输出网络电缆。这两种接头还配有开关，可选择网络偏流和终端。图 6-1 显示了电缆接头的普通偏流和终端状况，将拨钮拨向一侧，电阻设置为“on”，而将拨钮拨向另一侧，则电阻设置为“off”，图中只显示了一个，若有多个也是这样设置。图 6-1 中拨钮在“off”一侧，因此终端电阻未接入电路。

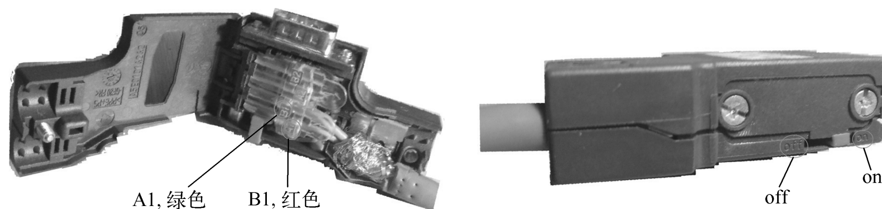


图 6-1　网络接头的终端电阻设置图

【关键点】西门子的专用 PROFIBUS 电缆中有两根线，一根为红色，上标有“B”，一根为绿色，上面标有“A”，这两根线只要与网络接头上相对应的“A”和“B”接线端子相连即可（如“A”线与“A”接线端相连）。网络接头直接插在 PLC 的 PORT 口上即可，不需要其他设备。注意：三菱的 FX 系列 PLC 的 RS－485 通信要加 RS－485 专用通信模块和终端电阻。

6.1.3　PLC 网络的术语解释

PLC 网络中的名词、术语很多，现将常用的予以介绍：

1）站（Station）：在 PLC 网络系统中，将可以进行数据通信、连接外部输入/输出的物理设备称为“站”。例如，由 PLC 组成的网络系统中，每台 PLC 可以是一个“站”。

2）主站（Master Station）：PLC 网络系统中进行数据链接的系统控制站，主站上设置了控制整个网络的参数，每个网络系统只有一个主站，主站号固定为“0”，站号实际就是 PLC 在网络中的地址。

3）从站（Slave Station）：PLC 网络系统中，除主站外，其他的站称为“从站”。

4）远程设备站（Remote Device Station）：PLC 网络系统中，能同时处理二进制位、字的从站。

5）本地站（Local Station）：PLC 网络系统中，带有 CPU 模块并可以与主站以及其他本地站进行循环传输的站。

6）站数（Number of Station）：PLC 网络系统中，所有物理设备（站）所占用的“内存站数”的总和。

7）网关（Gateway）：又称网间连接器、协议转换器。网关在传输层上用以实现网络互连，是最复杂的网络互连设备，仅用于两个高层协议不同的网络互连。网关的结构和路由器类似，不同的是互连层。网关既可以用于广域网互连，也可以用于局域网互连。网关是一种充当转换重任的计算机系统或设备。在使用不同的通信协议、数据格式或语言，甚至体系结构完全不同的两种系统之间，网关是一个翻译器。例如 AS－I 网络的信息要传送到由西门子 S7－200 系列 PLC 组成的 PPI 网络，就要通过 CP 243－2 通信模块进行转换，这个模块实际上就是网关。

8）中继器（Repeater）：用于网络信号放大、调整的网络互连设备，能有效延长网络的连接长度。例如，以太网的正常传送距离是 500m，经过中继器放大后，可传输 2500m。由于存在损耗，在线路上传输的信号功率会逐渐衰减，衰减到一定程度时将造成信号失真，因此会导致接收错误。中继器就是为解决这一问题而设计的。它完成物理线路的连接，对衰减的信号进行放大，保持与原数据相同。一般情况下，中继器的两端连接的是相同的媒体，但

有的中继器也可以完成不同媒体的转接工作。

9）网桥（Bridge）：网桥将两个相似的网络连接起来，并对网络数据的流通进行管理。网桥的功能在延长网络跨度上类似于中继器，然而它能提供智能化连接服务，即根据帧的终点地址处于哪一网段来进行转发和滤除。

10）路由器（Router，转发者）：所谓路由就是指通过相互连接的网络把信息从源地点移动到目标地点的活动。一般来说，在路由过程中，信息至少会经过一个或多个中间节点。路由器是互联网的主要节点设备。路由器通过路由决定数据的转发。转发策略称为路由选择（routing），这也是路由器名称的由来。作为不同网络之间互相连接的枢纽，路由器系统构成了基于 TCP/IP 的国际互联网络 Internet 的主体脉络，也可以说，路由器构成了 Internet 的骨架。它的处理速度是网络通信的主要瓶颈之一，它的可靠性则直接影响着网络互连的质量。因此，在园区网、地区网，乃至整个 Internet 研究领域中，路由器技术始终处于核心地位，其发展历程和方向，成为整个 Internet 研究的一个缩影。

11）交换机（Switch）：交换机是一种基于 MAC 地址识别，能完成封装转发数据包功能的网络设备。交换机可以"学习" MAC 地址，并把其存放在内部地址表中，通过在数据帧的始发者和目标接收者之间建立临时的交换路径，使数据帧直接由源地址到达目的地址。

交换机通过直通式、存储转发和碎片隔离三种方式进行交换。

交换机的传输模式有全双工、半双工、全双工/半双工自适应。

6.1.4 OSI 参考模型

通信网络的核心是 OSI（Open System Interconnection，开放式系统互连）参考模型。为了理解网络的操作方法，为创建和实现网络标准、设备和网络互连规划提供了一个框架，1984 年，国际标准化组织（ISO），提出了开放式系统互连的七层模型，即 OSI 模型。该模型自下而上分为：物理层、数据链接层、网络层、传输层、会话层、表示层和应用层。理解 OSI 参考模型比较难，但了解它，对掌握后续的以太网通信和 PROFIBUS 通信是很有帮助的。

OSI 的上三层通常称为应用层，用来处理用户接口、数据格式和应用程序的访问。下四层负责定义数据的物理传输介质和网络设备。OSI 参考模型定义了大多数协议栈共有的基本框架，信息在 OSI 模型中的流动形式如图 6-2 所示。

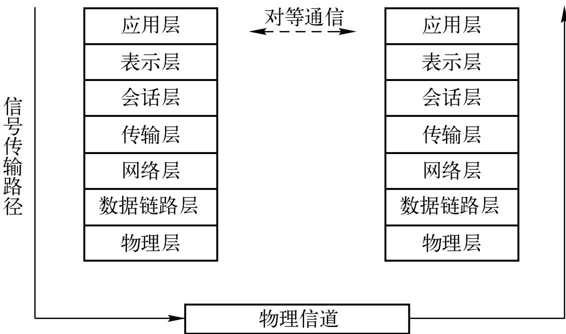


图 6-2　信息在 OSI 模型中的流动形式

1）物理层（Physical Layer）：定义了传输介质、连接器和信号发生器的类型，规定了物理连接的电气、机械特性，如电压、传输速率、传输距离等特性。建立、维护、断开物理连接。典型的物理层设备有集线器（HUB）和中继器等。

2）数据链路层（Data Link Layer）：确定传输站点物理地址以及将消息传送到协议栈，提供顺序控制和数据流向控制。建立逻辑连接、进行硬件地址寻址、差错校验等功能（由底层网络定义协议）。典型的数据链路层的设备有交换机和网桥等。

3）网络层（Network Layer）：进行逻辑地址寻址，实现不同网络之间的路径选择。协议为 ICMO IGMP IP（IPV4 IPV6）、ARP、RARP。典型的网络层设备是路由器。

4）传输层（Transport Layer）：定义传输数据的协议端口号，以及流控和差错校验。协议为 TCP、UDP。网关是互联网设备中最复杂的，它是传输层及以上层的设备。

5）会话层（Session Layer）：建立、管理、终止会话。

6）表示层（Presentation Layer）：数据的表示、安全、压缩。

7）应用层（Application）：网络服务与最终用户的一个接口。协议有：HTTP、FTP、TFTP SMTP、SNMP、DNS。

数据经过封装后通过物理介质传输到网络上，接收设备除去附加信息后，将数据上传到上层堆栈层。

各层的数据单位一般有各自特定的称谓。物理层的单位是比特（bit）；数据链路层的单位是帧（frame）；网络层的单位是分组（packet，有时也称包）；传输层的单位是数据报（datagram）或者段（segment）；会话层、表示层和应用层的单位是消息（message）。

6.1.5 工业通信网络结构

企业网是对工业企业计算机与控制网络的统称。企业网从结构上可以分为信息网络和控制网络两个层次，企业网的结构如图 6-3 所示。

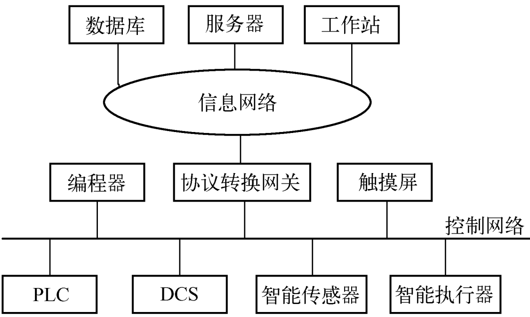


图 6-3 企业网的结构

信息网络是指用于企业内部的信息通信与管理的局域网。信息网络目前的主要应用是办公自动化。信息网络是接入互联网的，并且很多应用也是基于互联网技术的。

控制网络是指工业企业生产现场的通信网络。控制网络既可以是现场总线，也可以是工业以太网。控制网络主要实现现场设备之间、现场设备与控制器之间、现场设备与监控设备之间的通信。

网络化控制的功能模型是从功能的角度对基于网络的自动控制系统进行分层，简称

网络控制模型。网络控制模型分为现场设备层、监控层和管理层，如图 6-4 所示。

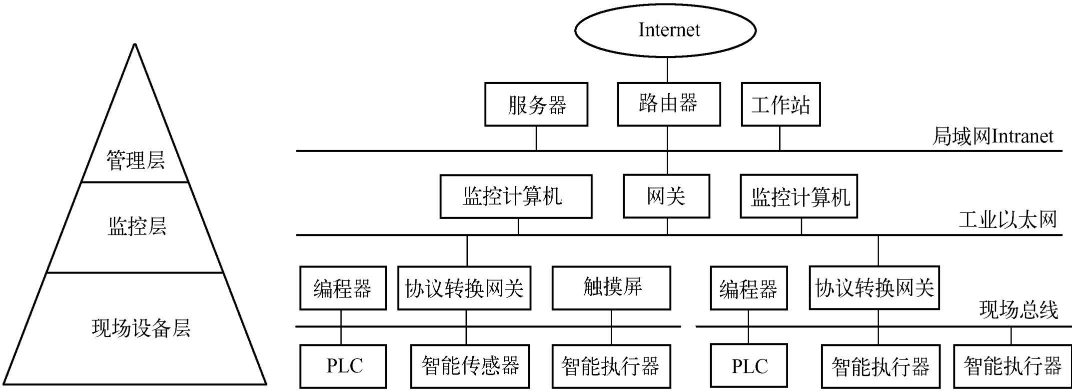


图 6-4　网络控制模型

1. 管理层

管理层为企业提供生产、管理和经营数据，通过数据化的方式优化企业资源，提高企业的管理水平。这个层中，IT 技术得到了广泛的应用，如 Internet 和 Intranet。

2. 监控层

监控层介于管理层和现场层之间。其主要功能是解决车间内各需要协调工作的不同工艺段之间的通信。监控层要求能传递大量的信息数据和少量控制信息，而且要求具备较强的实时性。这个层主要使用工业以太网。

3. 现场设备层

现场设备层处于工业网络的最底层，直接连接现场的各种设备，包括 I/O 设备、变频与驱动、传感器和变送器等，由于连接的设备千差万别，因此所使用的通信方式也比较复杂。又由于现场级通信网络直接连接现场设备，网络上传递的主要是控制信号，因此，对网络的实时性和确定性有很高的要求。

SIMATIC NET 中，现场级通信网络中主要使用 PROFIBUS。同时 SIMATIC NET 也支持 AS – Interface、EIB 等总线技术。

6.1.6　西门子通信网络技术说明

1. MPI 通信

MPI（Multi Point Interface，多点接口）协议，用于小范围、少点数的现场级通信。MPI 是为 S7/M7/C7 系统提供接口，用于编程设备的接口，也可用于在少数 CPU 间传递少量的数据。

2. PROFIBUS 通信

PROFIBUS 符合国际标准 IEC61158，是目前国际上通用的现场总线中 20 大现场总线之一，并以独特的技术特点、严格的认证规范、开放的标准和众多的厂家支持成为现场级通信网络的优秀解决方案，目前其全球网络节点已经突破 3000 万个。

从用户的角度看，PROFIBUS 提供三种通信协议类型：PROFIBUS – FMS、PROFIBUS – DP 和 PROFIBUS – PA。

1）PROFIBUS – FMS（Fieldbus Message Specification，现场总线报文规范）。用于系统级和车间级的不同供应商的自动化系统之间传输数据，处理单元级（PLC 和 PC）的多主站数据通信。目前，PROFIBUS – FMS 已经很少使用。

2）PROFIBUS – DP（Decentralized Periphery，分布式外部设备）。用于自动化系统中单元级控制设备与分布式 I/O（例如 ET 200）的通信。主站之间的通信为令牌方式，主站与从站之间为主从方式，以及这两种方式的混合。

3）PROFIBUS – PA（Process Automation，过程自动化）。用于过程自动化的现场传感器和执行器的低速数据传输，使用扩展的 PROFIBUS – DP 协议。

3. 工业以太网

工业以太网符合 IEEE802.3 国际标准，是功能强大的区域和单元网络，是目前工控界最为流行的网络通信技术之一。

4. 点对点连接

严格地说，点对点（Point – to – Point）连接并不是网络通信。但点对点连接可以通过串口连接模块实现数据交换，应用比较广泛。

5. AS – Interface

传感器/执行器接口用于自动化系统最底层的通信网络。它专门用来连接二进制的传感器和执行器，每个从站的最大数据量为 4 bit。

6.2 PPI 通信及其应用

6.2.1 PPI 通信基础

西门子的 S7 – 200 系列 PLC 可以支持 PPI 通信、MPI 通信（从站）、Modbus 通信、USS 通信、自由口协议通信、PROFIBUS – DP 现场总线通信（从站）、AS – I 通信和以太网通信等。

PPI 是一个主从协议，主站向从站发出请求，从站作出应答。从站不主动发出信息，而是等候主站向其发出请求或查询，要求应答。主站通过由 PPI 协议管理的共享连接与从站通信。PPI 不限制能够与任何一台从站通信的主站数目，但是无法在网络中安装 32 台以上的主站。

PPI 高级协议允许网络设备在设备之间建立逻辑连接。若使用 PPI 高级协议，每台设备可提供的连接数目有限。表 6–2 显示了 S7 – 200 提供的连接数目。PPI 协议目前还没有公开。

表 6–2　S7 – 200 提供的连接数目

模　　块	端　　口	波　特　率	连　　接
S7 – 200 CPU	端口 0	9.6 kbit/s、19.2 kbit/s 或 187.5 kbit/s	4 个
	端口 1	9.6 kbit/s、19.2 kbit/s 或 187.5 kbit/s	4 个
EM 277 模块		9.6 kbit/s 至 12 Mbit/s	每个模块 6 个

如果在用户程序中启用PPI主站模式，S7－200 CPU可在处于RUN（运行）模式时用做主站。启用PPI主站模式后，可以使用“网络读取”（NETR）或“网络写入”（NETW）指令从其他S7－200 CPU读取数据或向S7－200 CPU写入数据。可以使用PPI协议与所有的S7－200 CPU通信。如果与EM 277通信，必须启用“PPI高级协议”。

PPI主站的定义如下：

PLC用特殊寄存器的字节SMB30（对PORT0，端口0）和SMB130（对PORT1，端口1）定义通信口。控制位的定义格式如图6-5所示。

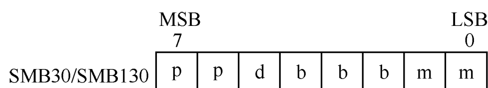


图6-5　控制位的定义格式

1）通信模式由控制字的最低的两位“mm”决定。

- mm＝00：PPI从站模式（默认值）。
- mm＝01：自由口模式。
- mm＝10：PPI主站模式。

所以，只要将SMB30或SMB130赋值为2#10，即可将通信口设置为PPI主站模式。

2）控制位的“pp”是奇偶校验选择。

- pp＝00：无校验。
- pp＝01：偶校验。
- pp＝10：无校验。
- pp＝11：奇校验。

3）控制位的“d”是每个字符的数据位选择。

- d＝0：每个字符8位。
- d＝1：每个字符7位。

4）控制位的“bbb”是波特率选择。

- bbb＝000：38400 bit/s。
- bbb＝001：19200 bit/s。
- bbb＝010：9600 bit/s。
- bbb＝011：4800 bit/s。
- bbb＝100：2400 bit/s。
- bbb＝101：1200 bit/s。
- bbb＝110：115200 bit/s。
- bbb＝111：57600 bit/s。

6.2.2　PPI通信的应用

PPI通信的实现比较简单，通常有两种方法：方法一是用STEP 7－Micro/WIN中的“指令向导”生成通信子程序，这种方法比较简单，适合初学者使用；方法二是用网络读/写指令编写通信程序，相对而言，要麻烦一些。以下用这两种方法，介绍两台PLC的

PPI 通信。

方法一——用指令向导

【例 6-1】 某设备的第一站和第二站上的控制器是 CPU 226CN，两个站组成一个 PPI 网络，其中，第一站的 PLC 为主站，第二站的 PLC 为从站。其工作任务是：当按下主站上的按钮 SB1 时，从站上的灯亮；当按下从站上的按钮 SB1 时，主站上的灯亮。请编写程序。

【解】

1. 主要软硬件配置

① 1 套 STEP 7 - Micro/WIN V4.0 SP9。

② 2 台 CPU 226CN。

③ 1 根 PROFIBUS 网络电缆（含两个网络总线连接器）。

④ 1 根 PC/PPI 电缆。

PROFIBUS 网络电缆、PPI 通信硬件配置、主站和从站接线分别如图 6-6、图 6-7、图 6-8 所示。



图 6-6　PROFIBUS 网络电缆

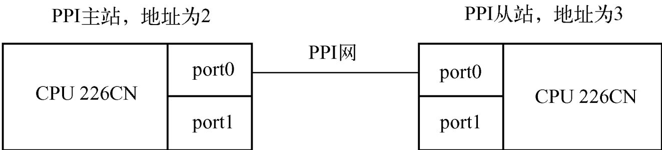


图 6-7　PPI 通信硬件配置图

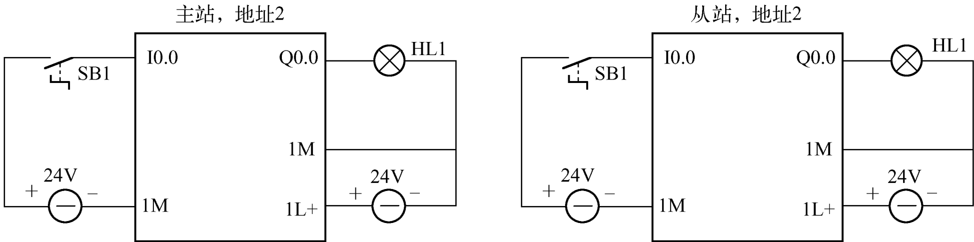


图 6-8　主站和从站接线图

2. 硬件配置过程

1）选择“NETR/NETW”。首先单击工具条中的“指令向导”按钮，弹出“指令向导”对话框，如图 6-9 所示，选中“NETR/NETW”选项，单击“下一步”按钮。

2）指定需要的网络操作数目。在图 6-10 所示的界面中设置需要进行多少网络读/写操作，由于本例有一个网络读取和一个网络写，故设为“2”即可，单击“下一步”按钮。

3）指定端口号和子程序名称。由于 CPU226 有 PORT0 和 PORT1 两个通信口，网络连接器插在哪个端口，配置时就选择哪个端口，子程序的名称可以不作更改，因此在图 6-11 所示的界面中，直接单击“下一步”按钮。

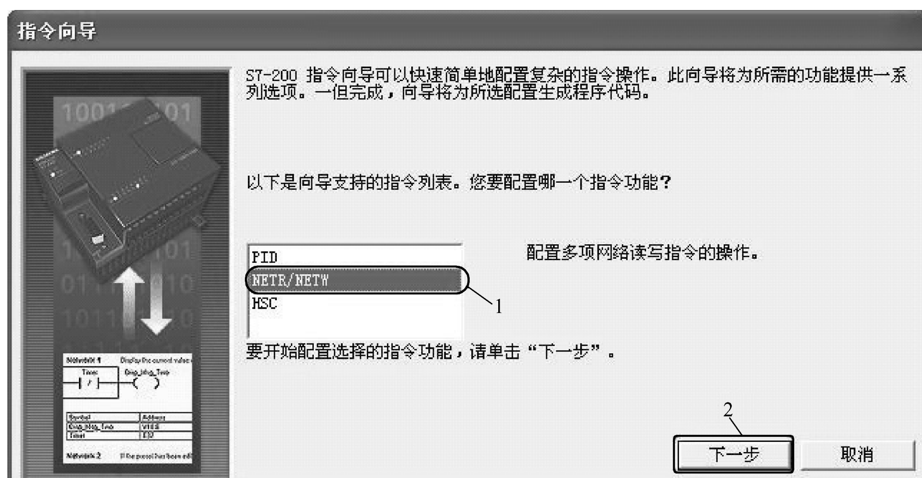


图 6-9　选择“NETR/NETW”

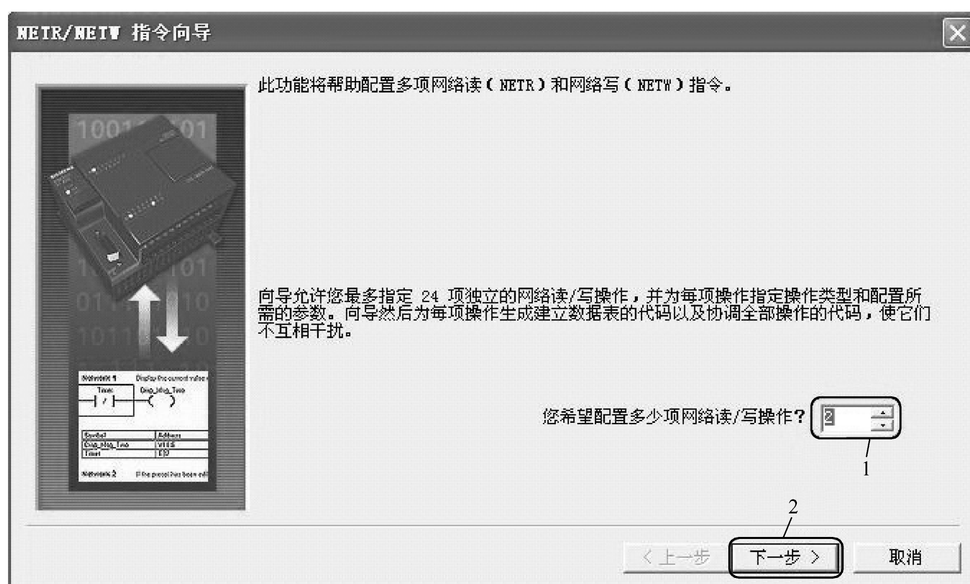


图 6-10　指定需要的网络读/写操作数目

4）指定网络操作。图 6-12 所示的界面相对比较复杂，需要设置 5 项参数。在图中的位置“1”，选择“NETR”（网络读），主站读取从站的信息；在位置“2”输入“1”，因为只有 1 个开关量信息；在位置“3”输入“3”，因为第三站的地址为“3”；位置“4”和位置“5”输入“VB1”，然后单击“下一项操作”按钮。

如图 6-13 所示，在图中的位置“1”，选择“NETW”（网络写），主站向从站发送信息；在位置“2”输入“1”，因为只有 1 个开关量信息；在位置“3”输入“3”，因为第三站的地址为“3”；位置“4”和位置“5”输入“VB0”，然后单击“下一步”按钮。

5）分配 V 存储区。接下来在图 6-14 所示的界面中分配系统要使用的存储区，通常使用默认值，然后单击“下一步”按钮。

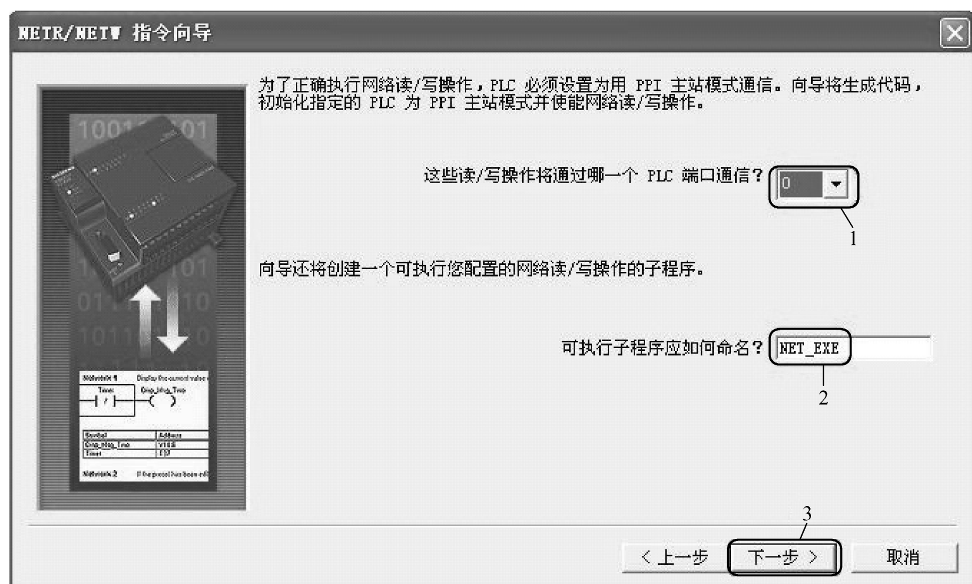


图 6-11　指定端口号和子程序名称

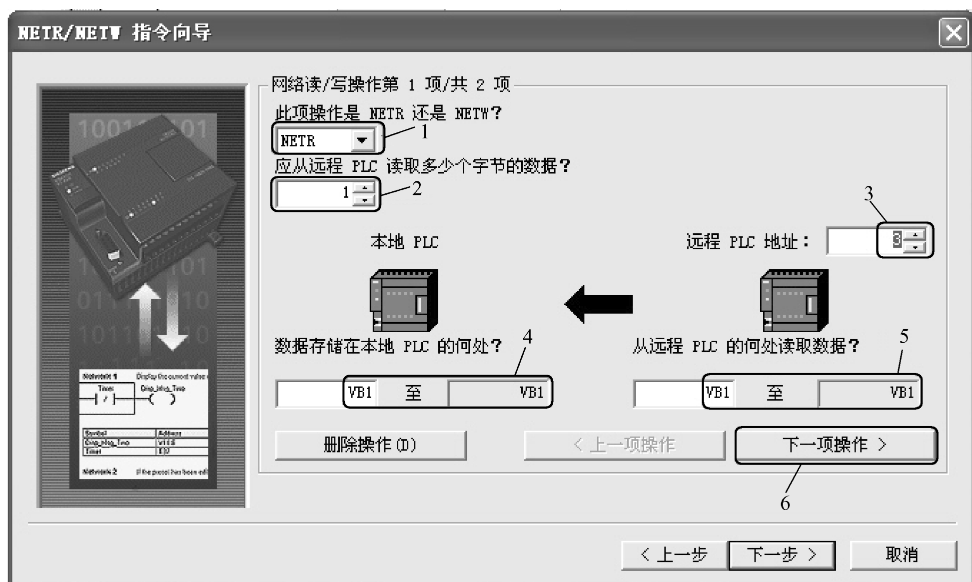


图 6-12　指定网络读操作

6）生成程序代码。最后单击“完成”按钮，如图 6-15 所示。至此通信子程序“NET_EXE”已经生成，在后面的程序中可以方便地进行调用。

3. 编写程序

通信子程序只在主站中调用，从站不调用通信子程序，从站只需要在指定的 V 存储单元中读/写相关的信息即可。主站和从站的程序如图 6-16 所示。

【关键点】 本例的主站站地址为“2”，在运行程序前，必须将从站的站地址设置成“3”（与图 6-12 中设置一致），此外，本例实际是将主站的 VB0 中数据传送到从站的 VB0

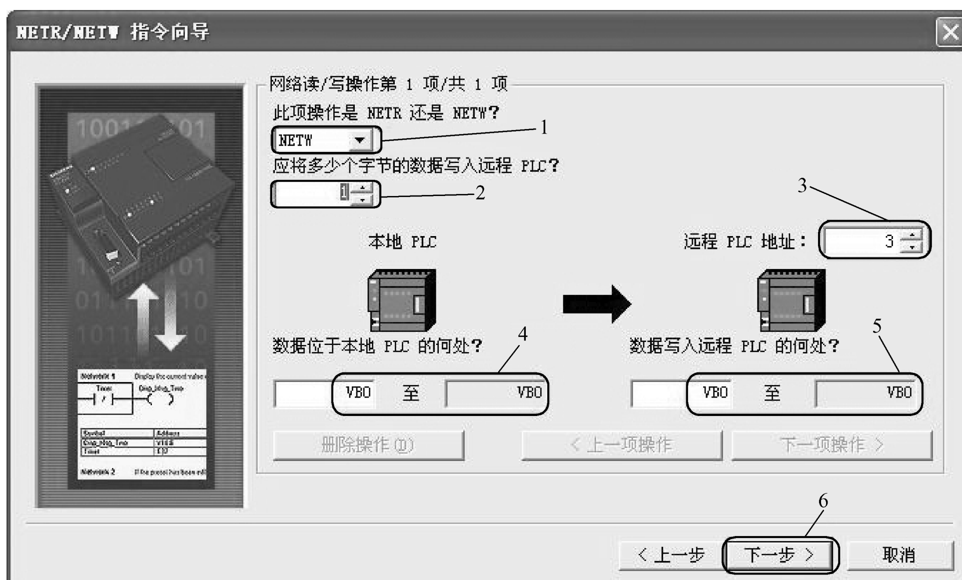


图 6-13　指定网络写操作

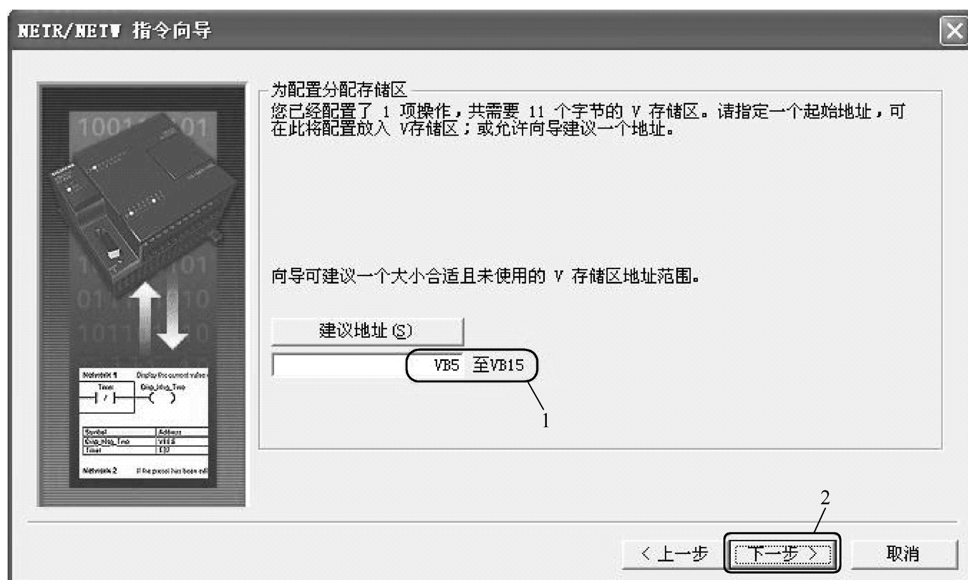


图 6-14　分配 V 存储区

中。此外，要注意站地址和站内地址的区别。主站和从站的波特率必须一致。一般而言，其他的通信方式也遵循这个原则，这点初学者很容易忽略。

方法二——用网络读/写指令

1. 网络读/写指令的格式

网络读取（NETR）指令，通过指定的端口（PORT）根据表格（TBL）定义从远程设备读取数据。NETR 指令可从远程站最多读取 16 字节信息。网络写入（NETW）指令通过指定的端口（PORT）根据表格（TBL）定义向远程设备写入数据。NETW 指令可向远程站最

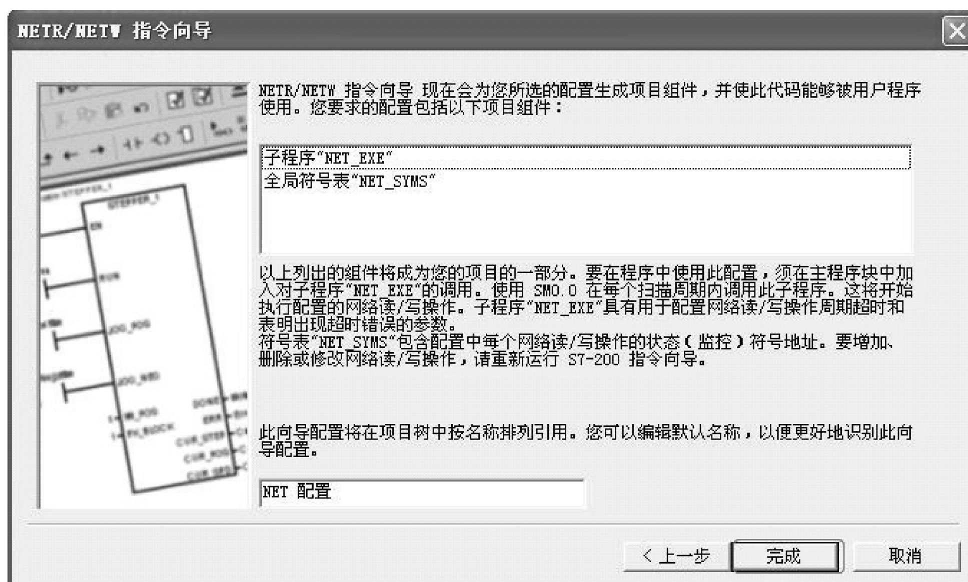


图 6-15　生成程序代码

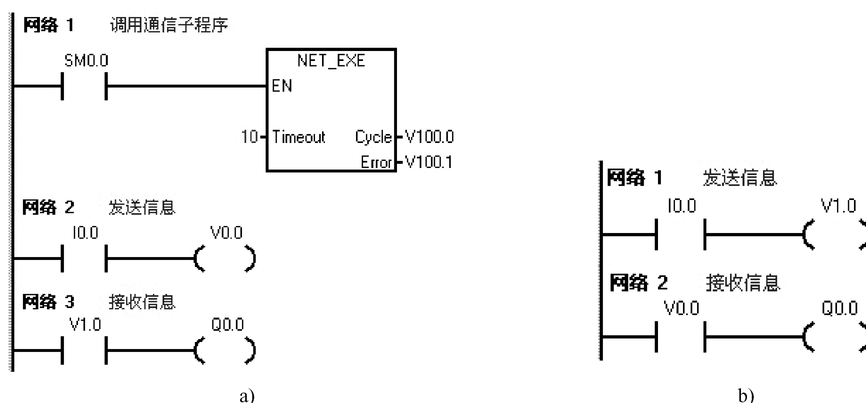


图 6-16　主站和从站程序

a）主站程序　b）从站程序

多写入 16 字节信息。可在程序中保持任意数目的 NETR/NETW 指令，但在任何时间最多只能有 8 条 NETR 和 NETW 指令被激活。例如，在特定 S7－200 PLC 中的同一时间可以有 4 条 NETR 和 4 条 NETW 指令（或者 2 条 NETR 和 6 条 NETW 指令）处于现用状态。网络读/写指令格式见表 6-3 所示。

表 6-3　网络读/写指令格式

LAD	参　数	说　明	数据类型
NETR EN　ENO TBL PORT	EN	使能端	BOOL
	TBL	读入的参数表的起始地址 VB	BYTE
	PORT	端口号，取值是 0、1	BYTE

（续）

LAD	参　　数	说　　明	数据类型
NETW EN　ENO TBL PORT	EN	使能端	BOOL
	TBL	写出的参数表的起始地址 VB	BYTE
	PORT	端口号，取值是 0、1	BYTE

2. 网络读/写指令的数据缓冲区

网络读/写指令具有相似的数据缓冲区，缓冲区以一个状态字起始。主站的数据缓冲区如图 6-17 所示。远程站的数据缓冲区如图 6-18 所示。

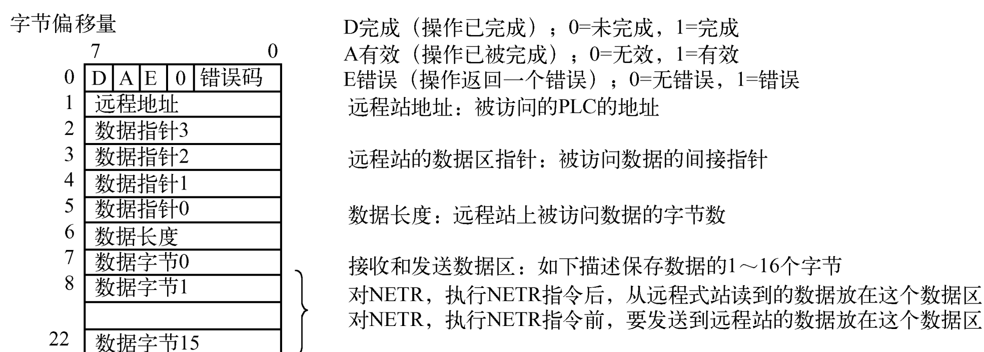


图 6-17　主站的数据缓冲区

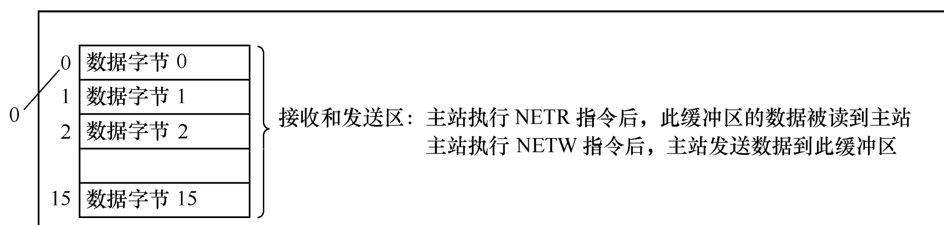


图 6-18　远程站的数据缓冲区

首先列出主站发送数据缓冲区和从站接收数据缓冲区，见表 6-4 和表 6-5。

表 6-4　主站发送数据缓冲区

VB200	状　态　字
VB201	从站的地址（3）
VD202	&VB400 从站的接收缓冲区地址
VB206	1（字节）
VB207	主站的 IB0

表 6-5　从站接收数据缓冲区

	起停信息
VB400	主站的 IB0

然后再列出主站接收数据缓冲区和从站发送数据缓冲区，见表 6-6 和表 6-7。

表 6-6　主站接收数据缓冲区

VB300	状　态　字
VB301	从站的地址（3）
VD302	&VB300 从站的发送缓冲区地址
VB306	1（字节）
VB307	从站的 IB0

表 6-7　从站发送数据缓冲区

	起 停 信 息
VB300	从站的 IB0

3. 编写程序

编写程序，如图 6-19 ~ 图 6-21 所示。

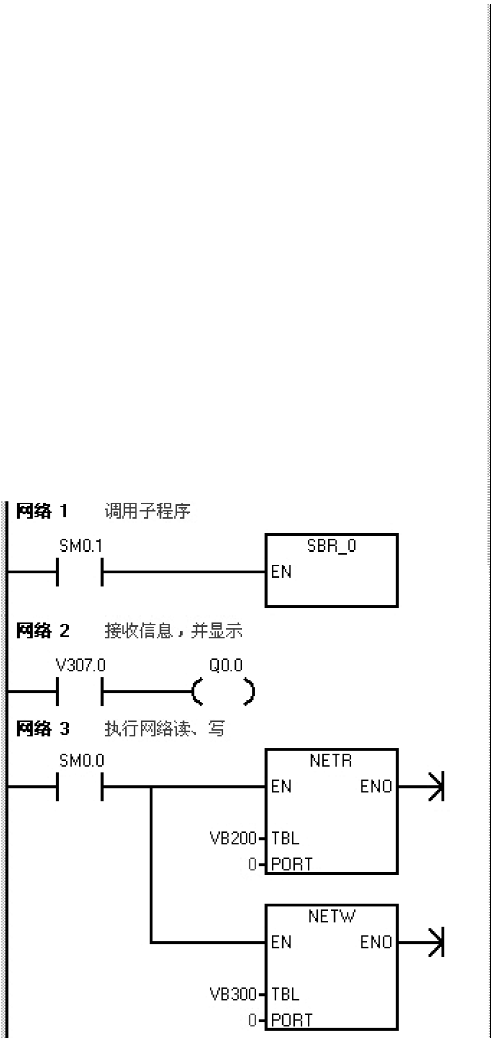


图 6-19　主站主程序

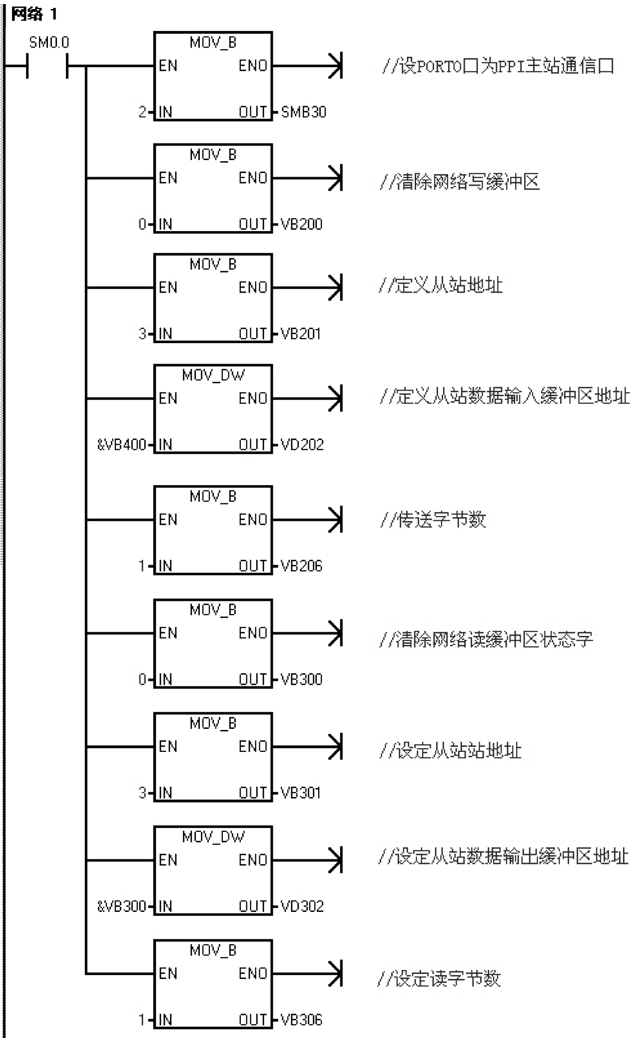


图 6-20　主站子程序

由此可见，用指令向导生成子程序进行 PPI 通信比使用网络读/写指令（NETR/NETW）要容易得多。

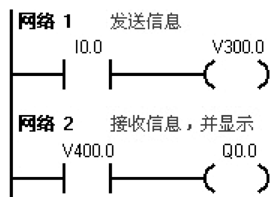


图 6-21　从站程序

6.3　S7-200 PLC 的 MPI 通信及应用

6.3.1　MPI 通信概述

MPI 网络可用于单元层，它是多点接口（MultiPoint Interface）的简称，是西门子公司开发的用于 PLC 之间通信的保密协议。MPI 通信是当通信速率要求不高、通信数据量不大时，可以采用的一种简单、经济的通信方式。

MPI 通信的主要优点是 CPU 可以同时与多种设备建立通信联系。也就是说，编程器、HMI 设备和其他的 PLC 可以连接在一起并同时运行。编程器通过 MPI 接口生成的网络还可以访问所连接硬件站上的所有智能模块。可同时连接的其他通信对象的数目取决于 CPU 的型号。例如，CPU 314 的最大连接数为 4，CPU 416 为 64。

MPI 接口的主要特性如下：

① RS-485 物理接口。

② 传输速率为 19.2 kbit/s 或 187.5 kbit/s 或 1.5 Mbit/s。

③ 最大连接距离为 50 m（2 个相邻节点之间），有两个中继器时为 1100 m，采用光纤和星型耦合器时为 23.8 km。

④ 采用 PROFIBUS 元件（电缆、连接器）。

MPI 通信有全局数据通信、无组态通信和组态通信，以下将只介绍无组态通信。

6.3.2　无组态连接 MPI 通信的应用

无组态连接 MPI 通信适合 S7-200、S7-300、S7-400 之间的通信，通过调用 SFC65、SFC66、SFC67、SFC68 来实现。无组态通信不能和全局数据方式混合使用。

无组态通信分为双边通信方式和单边通信方式。

单边无组态通信方式只在一方编写通信程序，即客户端和服务器端的访问模式。编写程序的一方为客户端，另一方为服务器端。当 S7-200/300/400 进行单边无组态通信时，S7-300/400 既可作为客户端，也可以作为服务器端，但 S7-200 只能作为服务器端。

【例 6-2】有两台设备，分别由一台 CPU 314C-2DP 和一台 CPU 226CN 控制，从设备 1 上的 CPU 314C-2DP 发出起停控制命令，设备 2 的 CPU 226CN 收到命令后，对设备 2 进行起停控制，同时设备 1 上的 CPU 314C-2DP 监控设备 2 的运行状态。

【解】

将设备 1 上的 CPU 314C-2DP 作为客户端，客户端的 MPI 地址为 2，将设备 2 上的 CPU

226CN 作为服务器端，服务器端的 MPI 地址为 3。

1. 主要软硬件配置

① 1 套 STEP 7 V5. 5CN。

② 1 台 CPU 314C－2DP。

③ 1 台 CPU 226CN。

④ 1 台 EM 277。

⑤ 1 根 PC/MPI 适配器（或者 CP 5611 卡）。

⑥ 1 根 MPI 电缆（含两个网络总线连接器）。

⑦ 1 套 STEP 7－Micro/WIN V4. 0 SP9。

MPI 通信硬件配置图如图 6-22 所示，PLC 接线图如图 6-23 所示。

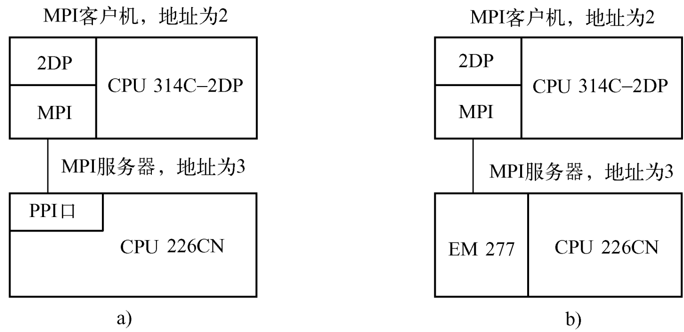


图 6-22　MPI 通信硬件配置图

a）方案 1　b）方案 2

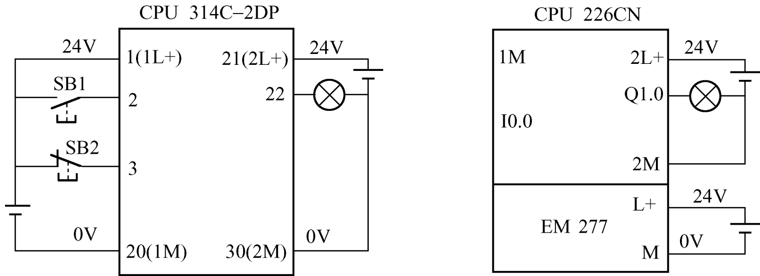


图 6-23　PLC 接线图

从图 6-22 可以看出 S7－200 系列 PLC 与 S7－300 系列 PLC 间的 MPI 通信有两种配置方案。方案 1 只要将 MPI 网络电缆（含两个网络总线连接器）连接在 S7－300 系列 PLC 的 MPI 接口和 S7－200 系列 PLC 的编程口上即可，而方案 2 却需要另加一个 EM 277 模块，显然成本多一些，但若 S7－200 系列 PLC 的编程接口不够用时，方案 2 是可以选择的配置方案。

2. 硬件组态

S7－200 系列 PLC 与 S7－300 系列 PLC 间的 MPI 通信只能采用无组态通信，无组态通信指通信无需组态，完成通信任务，只需要编写程序即可。只要用到 S7－300 系列 PLC，硬件组态还是不可缺少的，这点读者必须清楚。

1）新建工程并插入站点。新建工程，命名为“6-1”，再插入站点，重命名为“Master”，如图 6-24 所示，双击“硬件”图标，打开硬件组态界面。

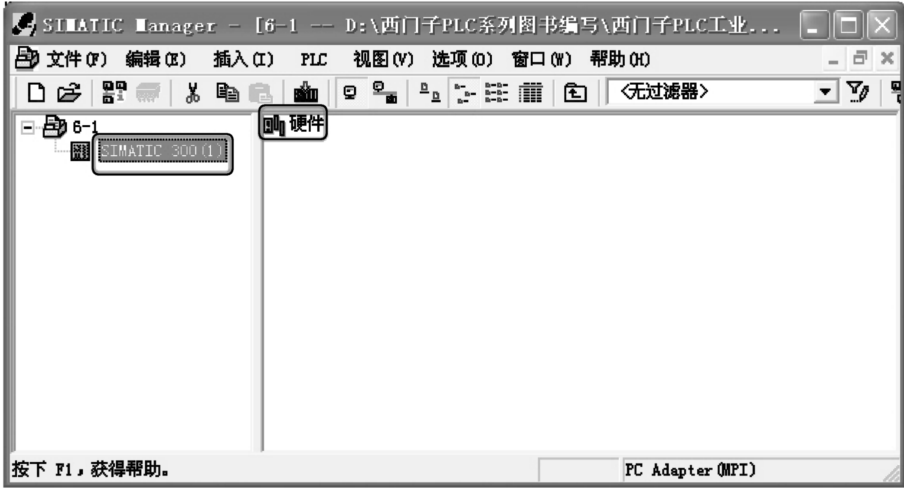


图 6-24　新建工程并插入站点

2）组态客户端硬件。先插入导轨，再插入 CPU 模块，如图 6-25 所示，双击“CPU 314C-2DP”，打开 MPI 通信参数设置界面，单击“属性”按钮，如图 6-26 所示。

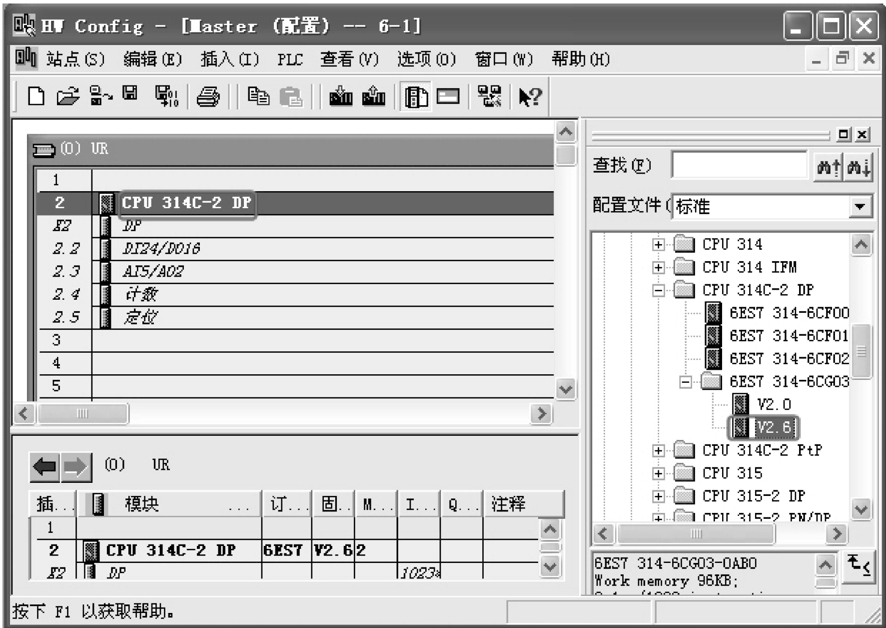


图 6-25　组态客户端硬件

3）设置客户端的 MPI 通信参数。先选定 MPI 的通信波特率为 187.5 kbit/s，再选定客户端的 MPI 地址为“2”，然后单击“确定”按钮，如图 6-27 所示。最后编译保存和下载硬件组态。

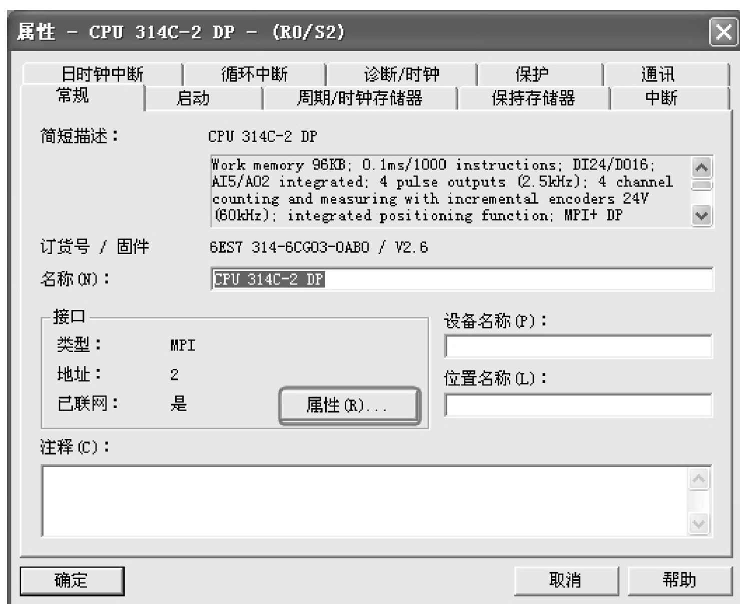


图 6-26　打开 MPI 通信参数设置界面



图 6-27　设置客户端的 MPI 通信参数

4）打开系统块。完成以上步骤后，S7－300 的硬件组态完成，但还必须设置 S7－200 的通信参数。先打开 STEP 7－Micro/WIN，选定工具条中的“系统块”按钮，并双击之，如图 6-28 所示。

5）设置服务器端的 MPI 通信参数。先将用于 MPI 通信的接口（本例为 port0）的地址设置成“3”，一定不能设定为“2”，再将波特率设定为“187.5 kbit/s”，这个数值与S7－300 的波特率必须相等，最后单击“确认”按钮，如图 6-29 所示。这一步不少初学者容易忽略，其实这一步非常关键，因为各站的波特率必须相等，这是一个基本原则。系统块设置完成后，还要将其下载到 S7－200 中，否则通信是不能建立的。



图 6-28　打开系统块

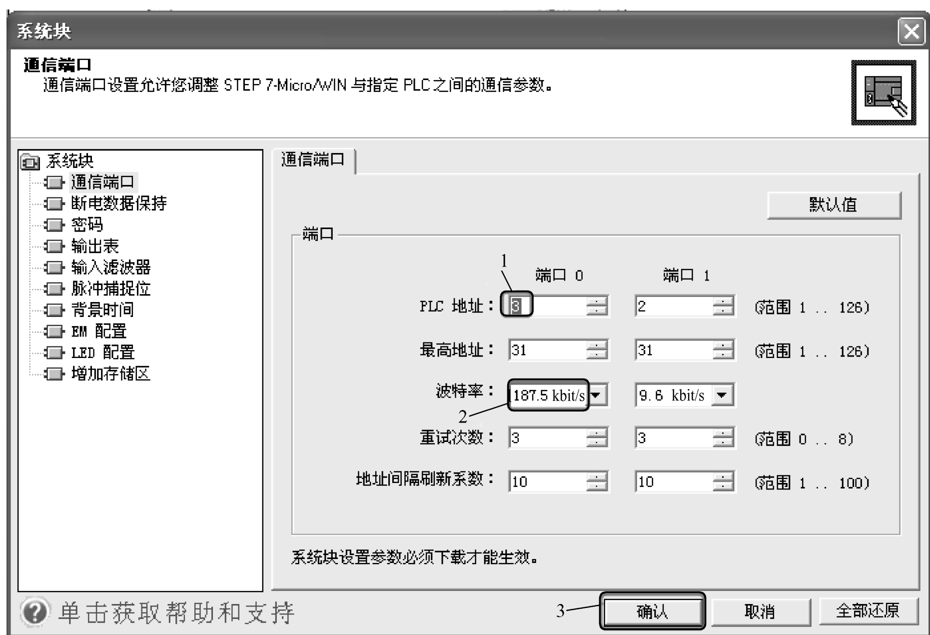


图 6-29　设置服务器端的 MPI 通信参数

【关键点】硬件组态时，必须将 S7-200 和 S7-300 的波特率设置值为相等，此外 S7-300 的硬件组态和 S7-200 的系统块必须下载到相应的 PLC 中才能起作用。

3. 相关指令介绍

X_PUT（SFC68）是发送数据的指令，通过 SFC68（X_PUT），将数据写入不在同一个本地 S7 站中的通信伙伴。在通信伙伴上没有相应系统功能块。在通过 REQ = 1 调用 SFC68 之后，激活写作业。此后，可以继续调用 SFC68，直到 BUSY = 0 指示接收到应答为止。

必须要确保由 SD 参数（在发送 CPU 上）定义的发送区和由 VAR_ADDR 参数（在通信

伙伴上）定义的接收区长度相同。SD 的数据类型还必须和 VAR_ADDR 的数据类型相匹配。X_PUT（SFC68）指令的输入和输出的含义见表 6-8。

表 6-8　X_PUT（SFC68）指令的输入和输出含义

LAD	输入/输出	说　明	数据类型
"X_PUT" EN　ENO REQ　RET_VAL CONT　BUSY DEST_ID VAR_ADDR SD	EN	使能	BOOL
	REQ	发送请求	BOOL
	CONT	作业结束之后是否"继续"保持与对方的连接	BOOL
	DEST_ID	对方的 MPI 地址	WORD
	VAR_ADDR	对方接收的数据存储区	ANY
	SD	本机的要发送的数据区	ANY
	RET_VAL	返回数值（如错误值）	INT
	BUSY	发送是否完成	BOOL

X_GET（SFC67）是接收数据的指令，通过 SFC67（X_GET），可以从本地 S7 站以外的通信伙伴中读取数据。在通信伙伴上没有相应系统功能块。在通过 REQ＝1 调用 SFC67 之后，激活读作业。此后，可以继续调用 SFC67，直到 BUSY＝0 指示数据接收为止。然后，RET_VAL 便包含了以字节为单位的、已接收的数据块的长度。

【关键点】 ANY 数据类型是指数据类型是任意的，即可以是字节、字、双字等，但接收端和发送端必须是相同的。

必须要确保由 RD 参数定义的接收区（在接收 CPU 上）至少和由 VAR_ADDR 参数定义的要读取的区域（在通信伙伴上）一样大。RD 的数据类型还必须和 VAR_ADDR 的数据类型相匹配。X_GET（SFC67）指令的输入和输出的含义见表 6-9。

表 6-9　X_GET（SFC67）指令的输入和输出含义

LAD	输入/输出	说　明	数据类型
"X_GET" EN　ENO REQ　RET_VAL CONT　BUSY DEST_ID　RD VAR_ADDR	EN	使能	BOOL
	REQ	接收请求	BOOL
	CONT	作业结束之后是否"继续"保持与对方的连接	BOOL
	DEST_ID	对方的 MPI 地址	WORD
	VAR_ADDR	对方的数据区	ANY
	RD	读取到本机的数据区	ANY
	RET_VAL	返回数值（如错误值）	INT
	BUSY	接收是否完成	BOOL

4. 程序编写

X_PUT（SFC68）发送数据的指令和 X_GET（SFC67）接收数据的指令是系统功能，也就是系统预先定义的功能，只要将"库"展开，再展开"Standard library（标准库）"，选定"X_PUT"或者"X_GET"，再双击之，"X_PUT"或者"X_GET"就自动在网络中指定的位置弹出，如图 6-30 所示。

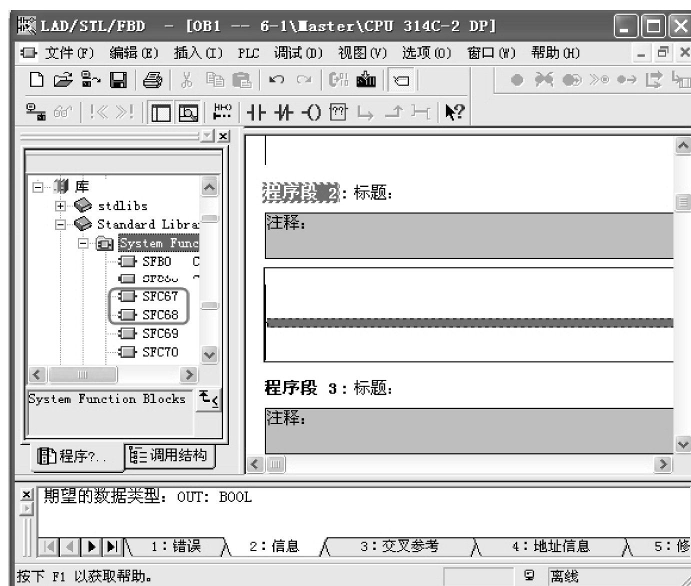
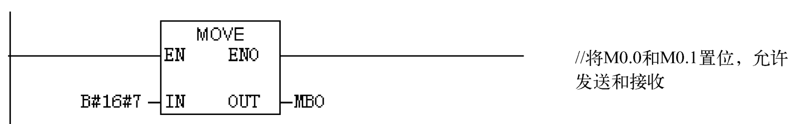


图 6-30 X_PUT 和 X_GET 指令的位置

客户端的程序如图 6-31 所示，服务器端并不需要编写程序。





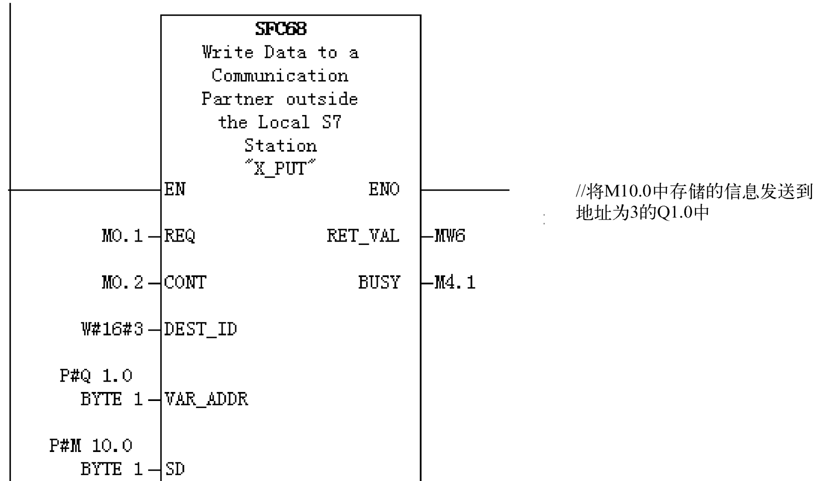


图 6-31 客户端程序

程序段 4：标题：

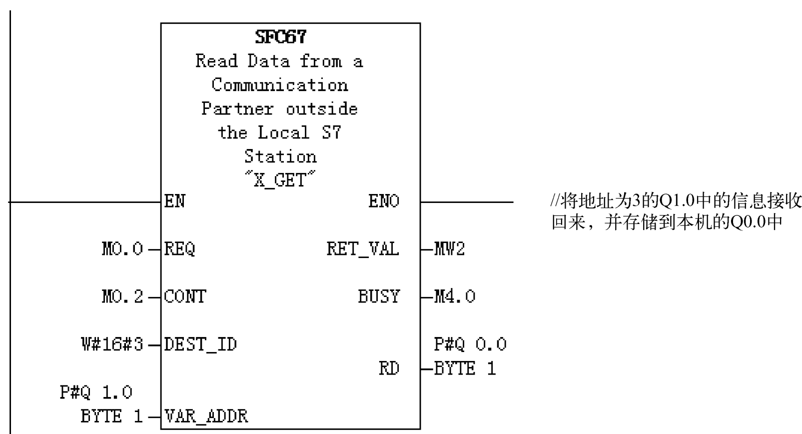


图 6-31　客户端程序（续）

【关键点】本例客户端地址为“2”，服务器端的地址为“3”，因此硬件配置采用方案 1 时，必须将“PPI 口”的地址设定为“3”。而采用方案 2 时，必须将 EM 277 的地址设定为“3”，设定完成后，还要将 EM 277 断电，新设定的地址才能起作用，方案 2 不用设置波特率。指令“X_PUT”的参数 SD 和 VAR_ADDR 的数据类型可以据实际情况确定，但在同一程序中数据类型必须一致。

6.4　自由口通信

6.4.1　自由口通信概述

S7-200 PLC 的自由口通信是基于 RS-485 通信基础的半双工通信，西门子 S7-200 系列 PLC 拥有自由口通信功能，顾名思义，就是没有标准的通信协议，用户可以自己规定协议。第三方设备大多支持 RS-485 串口通信，西门子 S7-200 系列 PLC 可以通过自由口通信模式控制串口通信。最简单的使用案例就是只用发送指令（XMT）向打印机或者变频器等第三方设备发送信息。不管任何情况，都通过 S7-200 系列 PLC 编写程序实现。

自由口通信的核心就是发送（XMT）和接收（RCV）两条指令，以及相应的特殊寄存器控制。由于 S7-200 CPU 通信端口是 RS-485 半双工通信口，因此发送和接收不能同时处于激活状态。RS-485 半双工通信串行字符通信的格式可以包括一个起始位、7 或 8 位字符（数据字节）、一个奇/偶校验位（或者没有校验位）、一个停止位。

自由口通信速波特率可以设置为 1200 bit/s、2400 bit/s、4800 bit/s、9600 bit/s、19200 bit/s、38400 bit/s、57600 bit/s 或 115200 bit/s。凡是符合这些格式的串行通信设备，理论上都可以和 S7-200 CPU 通信。自由口模式可以灵活应用。STEP 7-Micro/WIN 的两个指令库（USS 和 Modbus RTU）就是使用自由口模式编程实现的。

S7-200 CPU 使用 SMB30（对于 Port0）和 SMB130（对于 Port1）定义通信口的工作模式，控制字节的定义如图 6-5 所示。

1. 发送指令

以字节为单位，XMT 向指定通信口发送一串数据字符，要发送的字符以数据缓冲区指定，一次发送的字符最多为 255 个。

发送完成后，会产生一个中断事件，对于 Port0 口为中断事件 9，而对于 Port1 口为中断事件 26。当然也可以不通过中断，而通过监控 SM4.5（对于 Port0 口）或者 SM4.6（对于 Port1 口）的状态来判断发送是否完成，如果状态为 1，说明完成。XMT 指令缓冲区格式见表 6-10。

表 6-10　XMT 指令缓冲区格式

序　　号	字节编号	内　　容
1	T+0	发送字节的个数
2	T+1	数据字节
3	T+2	数据字节
	…	…
256	T+255	数据字节

2. 接收指令

以字节为单位，RCV 通过指定通信口接收一串数据字符，接收的字符保存在指定的数据缓冲区，一次接收的字符最多为 255 个。

接收完成后，会产生一个中断事件，对于 Port0 口为中断事件 23，而对于 Port1 口为中断事件 24。当然也可以不通过中断，而通过监控 SMB86（对于 Port0 口）或者 SMB186（对于 Port1 口）的状态来判断发送是否完成，如果状态为非 0，说明完成。SMB86 和 SMB186 含义见表 6-11，SMB87 和 SMB187 含义见表 6-12。

表 6-11　SMB86 和 SMB186 含义

对于 Port0 口	对于 Port1 口	控制字节各位的含义
SM86.0	SM186.0	1 说明奇偶校验错误而终止接收
SM86.1	SM186.1	1 说明接收字符超长而终止接收
SM86.2	SM186.2	1 说明接收超时而终止接收
SM86.3	SM186.3	0
SM86.4	SM186.4	0
SM86.5	SM186.5	1 说明是正常收到结束字符
SM86.6	SM186.6	1 说明输入参数错误或者缺少起始和终止条件而结束接收
SM86.7	SM186.7	1 说明用户通过禁止命令结束接收

表 6-12　SMB87 和 SMB187 含义

对于 Port0 口	对于 Port1 口	控制字节各位的含义
SM87.0	SM187.0	0
SM87.1	SM187.1	1 使用中断条件，0 不使用中断条件
SM87.2	SM187.2	1 使用 SM92 或者 SM192 时间段结束接收 0 不使用 SM92 或者 SM192 时间段结束接收
SM87.3	SM187.3	1 定时器是信息定时器，0 定时器是内部字符定时器

（续）

对于 Port0 口	对于 Port1 口	控制字节各位的含义
SM87.4	SM187.4	1 使用 SM90 或者 SM190 检测空闲状态 0 不使用 SM90 或者 SM190 检测空闲状态
SM87.5	SM187.5	1 使用 SM89 或者 SM189 终止符检测终止信息 0 不使用 SM89 或者 SM189 终止符检测终止信息
SM87.6	SM187.6	1 使用 SM88 或者 SM188 起始符检测起始信息 0 不使用 SM88 或者 SM188 起始符检测起始信息
SM87.7	SM187.7	0 禁止接收，1 允许接收

与自由口通信相关的其他重要特殊控制字/字节见表 6-13。

表 6-13　其他重要特殊控制字/字节

对于 Port0 口	对于 Port1 口	控制字节或者控制字的含义
SMB88	SMB188	消息字符的开始
SMB89	SMB189	消息字符的结束
SMW90	SMW190	空闲线时间段，按毫秒设定。空闲线时间用完后接收的第一个字符是新消息的开始
SMW92	SMW192	中间字符/消息定时器溢出值，按毫秒设定。如果超过这个时间段，则终止接收消息
SMW94	SMW194	要接收的最大字符数（1～255B）。此范围必须设置为期望的最大缓冲区大小，即使不使用字符计数消息终端

RCV 指令缓冲区格式见表 6-14。

表 6-14　RCV 指令缓冲区格式

序　　号	字节编号	内　　容
1	T+0	接收字节的个数
2	T+1	起始字符（如果有）
3	T+2	数据字节
4		数据字节
	…	…
256	T+255	结束字符（如果有）

6.4.2　S7－200 系列 PLC 自由口通信的应用

以两台 S7－200 CPU 之间的自由口通信为例介绍 S7－200 系列 PLC 之间的自由口通信的编程实施方法。

【例 6-3】有两台设备，控制器都是 CPU 226CN，两者之间为自由口通信，要求实现设备 1 对设备 1 和 2 的电动机，同时进行起停控制，请设计方案，编写程序。

【解】

1. 主要软硬件配置

① 1 套 STEP 7－Micro/WIN V4.0 SP9。

② 2 台 CPU 226CN。

③ 1 根 PROFIBUS 网络电缆（含 2 个网络总线连接器）。

④ 1 根 PC/PPI 电缆。

自由口通信硬件配置如图 6-32 所示，两台 CPU 的接线如图 6-33 所示。

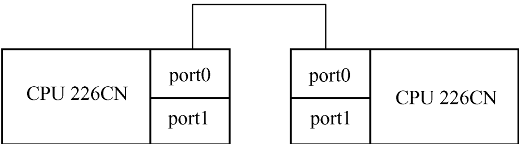


图 6-32　自由口通信硬件配置图

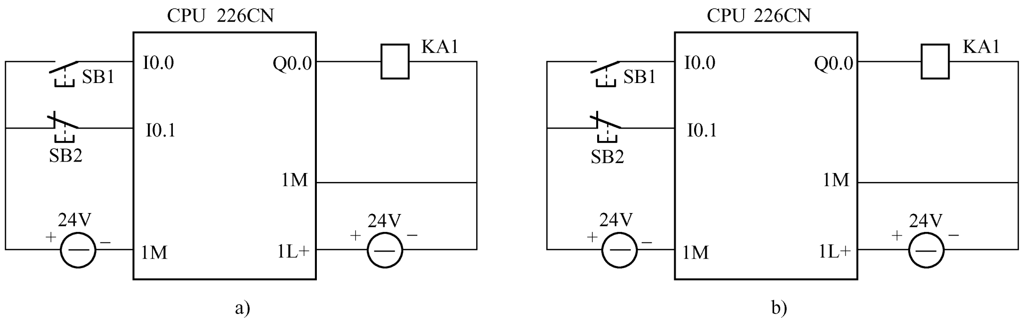


图 6-33　接线图

a）站 1　b）站 2

【关键点】 自由口通信的通信线缆最好使用 PROFIBUS 网络电缆和网络总线连接器，若要求不高，为了节省开支可购买市场上的 DB9 型接插件，再将两个接插件的 3 和 8 角对连即可，如图 6-34 所示。

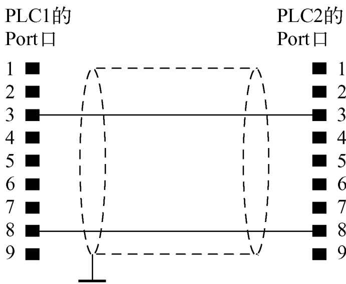


图 6-34　自由口通信连线的另一种方案

2. 编写设备 1 的程序

设备 1 的主程序如图 6-35。设备 1 的中断程序 0 如图 6-36。设备 1 的中断程序 1 如图 6-37。

3. 编写设备 2 的程序

设备 2 的主程序如图 6-38。设备 2 的中断程序 0 如图 6-39。

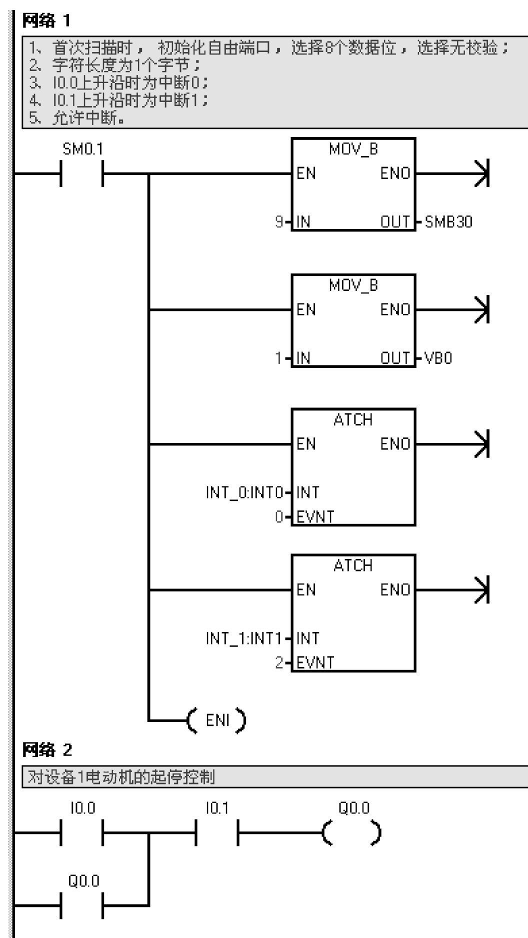


图 6-35　设备 1 主程序

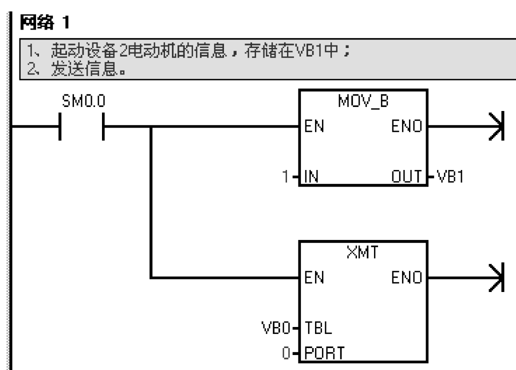


图 6-36　设备 1 中断程序 0



图 6-37　设备 1 中断程序 1

【例 6-4】用另一种方法解决例 6-3。

（1）设备 1 程序

设备 1 的主程序如图 6-40 所示。设备 1 的子程序如图 6-41 所示。设备 1 的中断程序如图 6-42 所示。

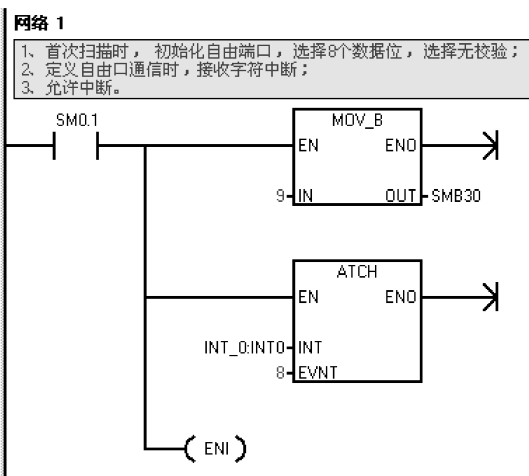


图 6–38　设备 2 主程序

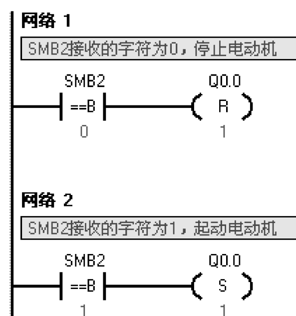


图 6–39　设备 2 中断程序 0

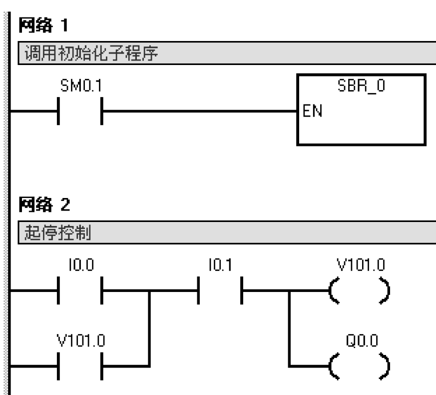


图 6–40　设备 1 主程序

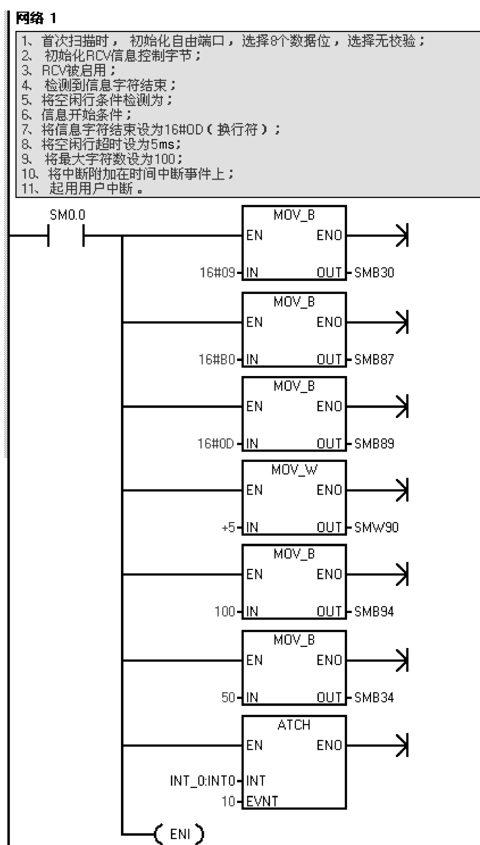


图 6–41　设备 2 子程序

（2）设备 2 程序

设备 2 的主程序如图 6–43 所示。设备 2 的中断程序如图 6–44 所示。

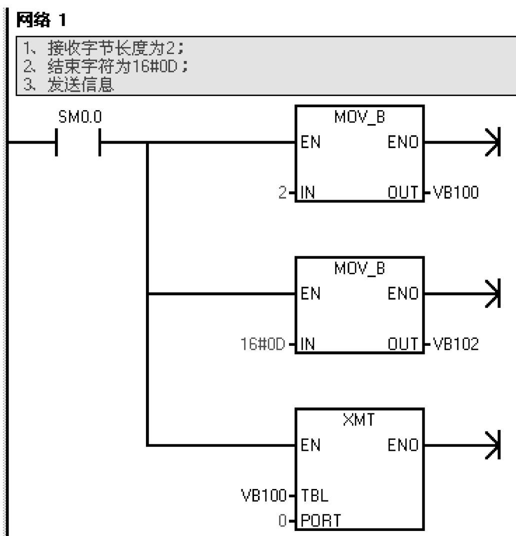


图 6-42　设备 1 中断程序

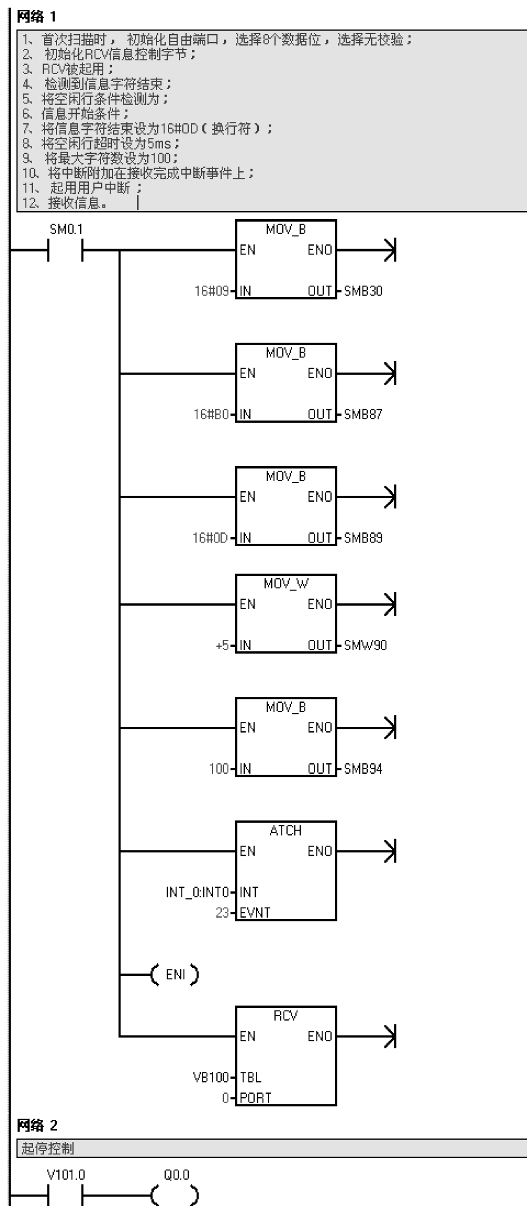


图 6-43　设备 2 主程序

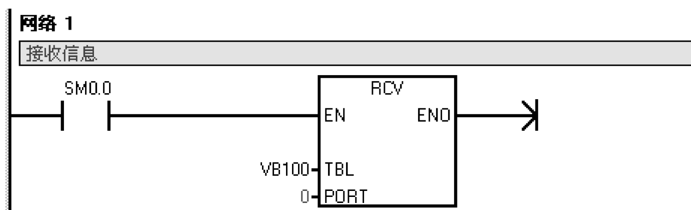


图 6-44　设备 2 中断程序

6.4.3　S7-200 PLC 与个人计算机（自编程序）的自由口通信

【例 6-5】用 Visual Basic 编写程序实现个人计算机和 CPU 226CN 的自由口通信，并显示 CPU 226CN 的 Q1.0～Q1.2 状态以及 QB0、QB1 的数值。

【解】

软硬件配置与例 6-4 相同。

将 CPU 226CN 作为主站，计算机作为从站。

1. 编写 CPU 226CN 的程序

CPU 226CN 中的程序如图 6-45 所示。

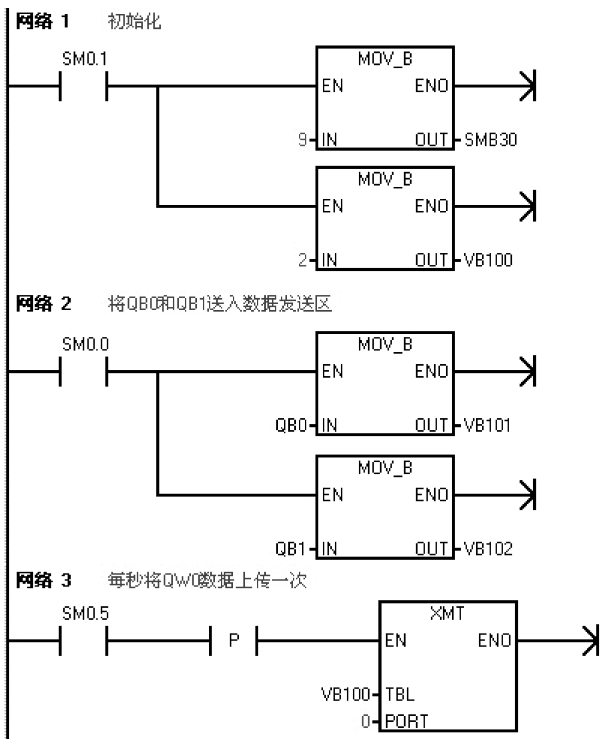


图 6-45　CPU 226CN 的程序

2. 编写计算机的程序

计算机中的程序用 Visual Basic 编写，程序运行界面如图 6-46 所示。

```
Option Explicit
Dim p( ) As Byte
Dim a
Private Sub Form_Load( )
    MSComm1.PortOpen = True           '打开串口
    MSComm1.InputMode = 1             '读入字节
    MSComm1.RThreshold = 2            '最少读入字节数
End Sub
```

```
Private Sub MSComm1_OnComm( )
    Select Case MSComm1. CommEvent
    Case comEvReceive
      p = MSComm1. Input                        '读入字节到数组 p(0)和 p(1)
      Text1 = p(1)
      Text2 = p(0)
      a = Val(Text1)
      Select Case a
      Case 0
          Shape1. BackColor = vbBlack           '当 QB0 = 0 时,三盏灯都不亮
          Shape2. BackColor = vbBlack
          Shape3. BackColor = vbBlack
      Case 1
          Shape1. BorderColor = vbRed           '当 QB0 = 1 时,第一盏灯亮
          Shape2. BackColor = vbBlack
          Shape3. BackColor = vbBlack
      Case 2
          Shape1. BackColor = vbBlack           '当 QB0 = 2 时,第二盏灯亮
          Shape2. BackColor = vbRed
          Shape3. BackColor = vbBlack
      Case 3
          Shape1. BackColor = vbRed             '当 QB0 = 3 时,第一、二盏灯亮
          Shape2. BackColor = vbRed
          Shape3. BackColor = vbBlack
      Case 4
          Shape1. BackColor = vbBlack           '当 QB0 = 4 时,第三盏灯亮
          Shape2. BackColor = vbBlack
          Shape3. BackColor = vbRed
      Case 5
          Shape1. BackColor = vbRed             '当 QB0 = 5 时,第一、三盏灯亮
          Shape2. BackColor = vbBlack
          Shape3. BackColor = vbRed
      Case 6
          Shape1. BackColor = vbBlack           '当 QB0 = 6 时,第二、三盏灯亮
          Shape2. BackColor = vbRed
          Shape3. BackColor = vbRed
      Case 7
          Shape1. BackColor = vbRed             '当 QB0 = 7 时,第一、二、三盏灯亮
          Shape2. BackColor = vbRed
          Shape3. BackColor = vbRed
      End Select
    End Select
End Sub
```

如图 6-46 所示，当 QB1 =5 时，Q1.0 和 Q1.2 两盏灯亮（显示为红色），而 Q1.1 灯灭（显示为黑色），书籍印刷可能无法表示，请读者运行本书附带的程序。

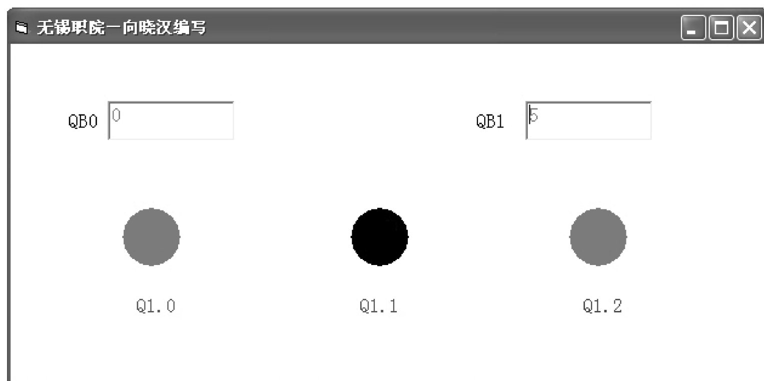


图 6-46　程序运行界面

6.4.4　S7 -200 PLC 与三菱 FX 系列 PLC 的自由口通信

除了 S7 -200 系列 PLC 之间可以进行自由口通信，S7 -200 系列 PLC 还可以与其他品牌的 PLC、变频器、仪表和打印机等进行通信。要完成通信，这些设备应有 RS -232C 或者 RS -485等形式的串口。西门子 S7 -200 与三菱的 FX 系列 PLC 通信时，采用自由口通信，但三菱公司称这种通信为“无协议通信”，内涵实际上是一样的。

以 CPU 226CN 与三菱 FX2N -32MR 自由口通信为例，讲解 S7 -200 系列 PLC 与其他品牌 PLC 或者之间的自由口通信。

【例 6-6】 有两台设备，设备 1 的控制器是 CPU 226CN，设备 2 的控制器是 FX2N -32MR，两者之间为自由口通信，实现设备 1 的 I0.0 起动设备 2 的电动机，设备 1 的 I0.1 停止设备 2 的电动机的转动，请设计解决方案。

【解】

1. 主要软硬件配置

① 1 套 STEP 7 -Micro/WIN V4.0 SP9 和 GX Developer 8.6。

② 1 台 CPU 226CN 和 1 台 FX2N -32MR。

③ 1 根屏蔽双绞电缆（含 1 个网络总线连接器）。

④ 1 台 FX2N -485-BD。

⑤ 1 根 PC/PPI 电缆。

两台 CPU 的接线如图 6-47 所示。

【关键点】 网络的正确接线至关重要，具体如下：

① CPU 226CN 的 PORT0 口可以进行自由口通信，其 9 针的接头中，1 号引脚接地，3 号引脚为 RXD +/TXD +（发送 +/接收 +）公用，8 号引脚为 RXD -/TXD -（发送 -/接收 -）公用。

② FX2N -32MR 的编程口不能进行自由口通信，因此本例配置了一块 FX2N -485-BD 模块，此模块可以进行双向 RS -485 通信（可以与两对双绞线相连），但由于 CPU 226CN 只能与一对双绞线相连，因此 FX2N -485-BD 模块的 RDA（接收 +）和 SDA（发送 +）短接，SDB（接收 -）和 RDB（发送 -）短接。

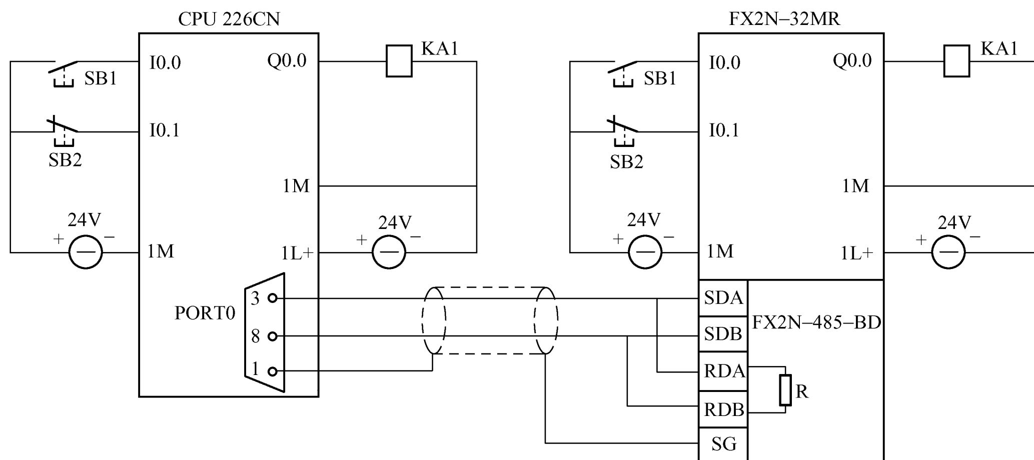


图 6–47　接线图

③ 由于本例采用的是 RS – 485 通信，所以两端需要接终端电阻，均为 110 Ω，CPU 226CN 端未画出（由于和 PORT0 相连的网络连接器自带终端电阻，有关内容在后面会详细讲解），若传输距离较近时，终端电阻可不接入。

2. 编写 CPU 226CN 的程序

CPU 226CN 中的主程序如图 6–48 所示，子程序如图 6–49 所示，中断程序如图 6–50 所示。

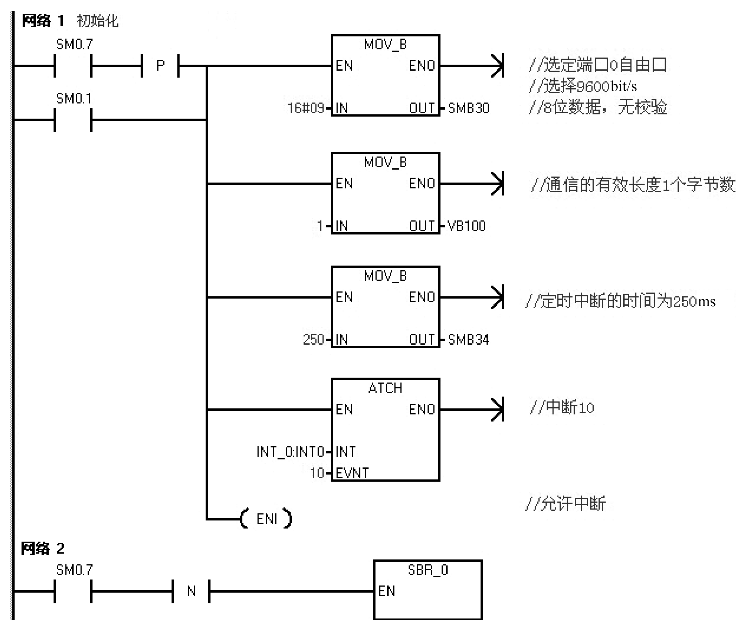


图 6–48　主程序

【关键点】自由口通信每次发送的消息最少是一个字节，本例中将起停消息存储在 VB101 的 V101.0 位发送出去。VB100 存放的是发送有效数据的字节数。

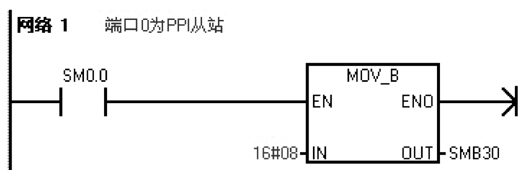


图 6-49　子程序

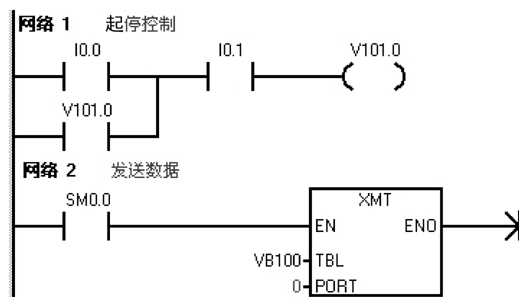


图 6-50　中断程序

3. 编写 FX2N－32MR 的程序

（1）无协议通信简介

1）RS 指令格式。RS 指令格式如图 6-51 所示。

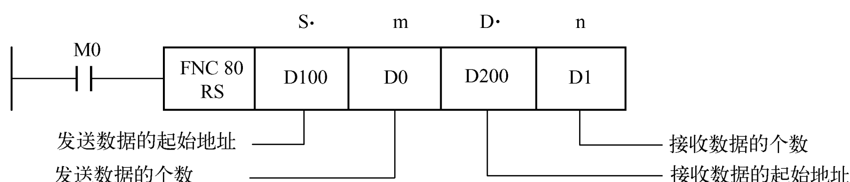


图 6-51　RS 指令格式

2）无协议通信中用到的软元件。无协议通信中用到的软元件见表 6-15。

表 6-15　无协议通信中用到的软元件

元件编号	名　　称	内　　容	属　　性
M8122	发送请求	置位后，开始发送	读/写
M8123	接收结束标志	接收结束后置位，此时不能再接收数据，须人工复位	读/写
M8161	8 位处理模式	在 16 位和 8 位数据之间切换接收和发送数据，为 ON 时为 8 位模式，为 OFF 时为 16 位模式	写

3）D8120 字的通信格式。D8120 的通信格式见表 6-16。

表 6-16　D8120 的通信格式

位　编　号	名　　称	内　　容	
		0（位 OFF）	1（位 ON）
b0	数据长度	7 位	8 位
b1b2	奇偶校验	b2，b1 （0，0）：无 （0，1）：奇校验（ODD） （1，1）：偶校验（EVEN）	

（续）

位 编 号	名 称	内 容	
		0（位 OFF）	1（位 ON）
b3	停止位	1 位	2 位
b4b5b6b7	波特率（bit/s）	b7，b6，b5，b4 （0，0，1，1）：300 （0，1，0，0）：600 （0，1，0，1）：1，200 （0，1，1，0）：2，400	b7，b6，b5，b4 （0，1，1，1）：4，800 （1，0，0，0）：9，600 （1，0，0，1）：19，200
b8	报头	无	有
b9	报尾	无	有
b10b11b12	控制线	无协议	b12，b11，b10 （0,0,0）：无 <RS－232C 接口> （0,0,1）：普通模式 <RS－232C 接口>（0,1,0）：相互链接模式 <RS－232C 接口>
		计算机链接	（0,1,1）：调制解调器模式 <RS－232C 接口> （1,1,1）：RS－485 通信 <RS－485/RS－422 接口>
b13	和校验	不附加	附加
b14	协议	无协议	专用协议
b15	控制顺序（CR 、LF）	不使用 CR，LF（格式 1）	使用 CR，LF（格式 4）

（2）编写程序

FX2N－32MR 中的程序如图 6-52 所示。

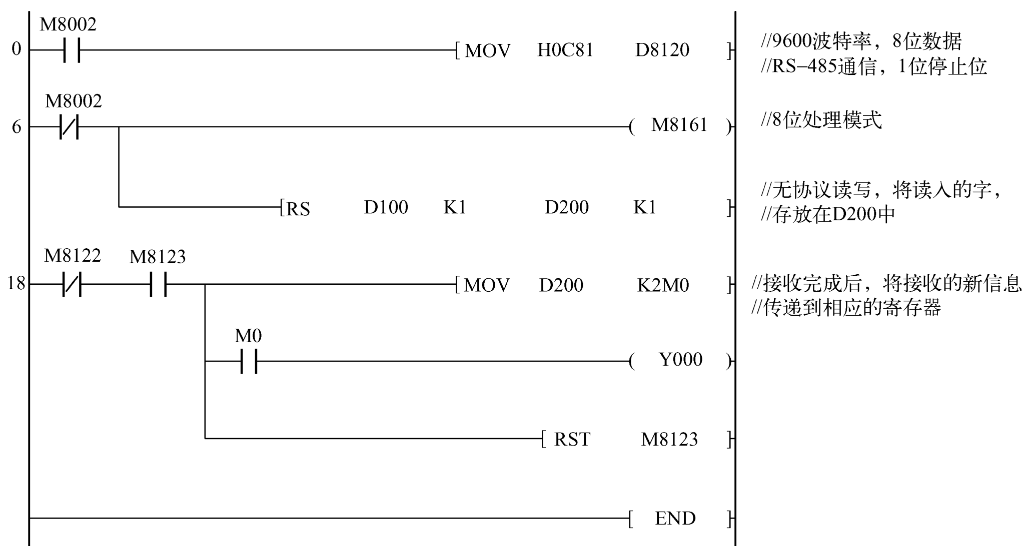


图 6-52 程序

实现不同品牌的 PLC 的通信，确实比较麻烦，要求读者对两种品牌的 PLC 的通信都比较熟悉。其中有两个关键点：一是读者一定要把通信线接对；二是与自由口（无协议）通信的相关指令必须要弄清楚，否则通信是很难建立的。

【关键点】以上的程序是单向传递数据，即数据只从 CPU226CN 传向 FX2N－32MR，因此程序相对而言比较简单，若要数据双向传递，则必须注意 RS－485 通信是半双工的，编写程序时要保证在同一时刻同一个站点只能接收或者发送数据。

6.5 PROFIBUS 现场总线及应用

6.5.1 现场总线的概念

1. 现场总线的诞生

现场总线是 20 世纪 80 年代中后期在工业控制中逐步发展起来的。随着微处理器技术的发展，其功能不断增强，而成本不断下降。计算机技术飞速发展，同时计算机网络技术也迅速发展起来了。计算机技术的发展为现场总线的诞生奠定了技术基础。

另一方面，智能仪表也出现在工业控制中。在原模拟仪表的基础上增加具有计算功能的微处理器芯片，在输出的 4～20 mA 直流信号上叠加了数字信号，使现场输入输出设备与控制器之间的模拟信号转变为数字信号。智能仪表的出现为现场总线的诞生奠定了应用基础。

2. 现场总线的概念

国际电工委员会（IEC）对现场总线（Fieldbus）的定义为：一种应用于生产现场，在现场设备之间、现场设备和控制装置之间实行双向、串行、多节点的数字通信网络。

现场总线的概念有广义与狭义之分。狭义的现场总线就是指基于 EIA485 的串行通信网络；广义的现场总线泛指用于工业现场的所有控制网络。广义的现场总线包括狭义现场总线和工业以太网。

工业以太网是用于工业现场的以太网，一般采用交换技术，即交换式以太网技术。工业以太网以 TCP/IP 为基础，与串行通信的技术体系是不同的。

在工业控制中，现场总线的概念因场合不同而不同。广义的“现场总线”，包括现场总线和工业以太网；而本书后续的章节中，现场总线的概念又是狭义的。读者应根据不同场合加以区别。

6.5.2 主流现场总线的简介

1984 年国际电工技术委员会/国际标准协会（IEC/ISA）就开始制定现场总线的标准，然而统一的标准至今仍未完成。很多公司推出其各自的现场总线技术，但彼此的开放性和互操作性难以统一。

经过 12 年的讨论，终于在 1999 年底通过了 IEC61158 现场总线标准，这个标准容纳了 8 种互不兼容的总线协议。后来又经过不断讨论和协商，在 2003 年 4 月，IEC61158 Ed. 3 现场总线标准第 3 版正式成为国际标准，确定了 10 种不同类型的现场总线为 IEC61158 现场总线，见表 6-17。2007 年 7 月，第四版现场总线增加到 20 种，新增的现场总线中，工业以太网就有 7 种。

表 6-17 IEC61158 的现场总线

类型编号	名　　称	发起的公司
Type 1	TS61158 现场总线	原来的技术报告
Type 2	ControlNet 和 Ethernet/IP 现场总线	美国 Rockwell 公司

（续）

类型编号	名　　称	发起的公司
Type 3	PROFIBUS 现场总线	德国 Siemens 公司
Type 4	P－NET 现场总线	丹麦 Process Data 公司
Type 5	FF HSE 现场总线	美国 Fisher Rosemount 公司
Type 6	SwiftNet 现场总线	美国波音公司
Type 7	World FIP 现场总线	法国 Alstom 公司
Type 8	INTERBUS 现场总线	德国 Phoenix Contact 公司
Type 9	FF H1 现场总线	现场总线基金会
Type 10	PROFINET 现场总线	德国 Siemens 公司

1. 基金会现场总线（Foundation Fieldbus，FF）

这是以美国 Fisher－Rosemount 公司为首的联合了横河、ABB、西门子、英维斯等 80 家公司制定的 ISP 协议和以 Honeywell 公司为首的联合欧洲等地 150 余家公司制定的 World FIP 协议于 1994 年 9 月合并的。该总线在过程自动化领域得到了广泛的应用，具有良好的发展前景。

2. CAN（Controller Area Network，控制器局域网）

最早由德国 BOSCH 公司推出，它广泛应用于离散控制领域，其总线规范已被国际标准化组织（ISO）制定为国际标准，得到了 Intel、Motorola、NEC 等公司的支持。CAN 协议分为两层：物理层和数据链路层。CAN 的信号传输采用短帧结构，传输时间短，具有自动关闭功能，具有较强的抗干扰能力。CAN 支持多种工作方式，并采用了非破坏性总线仲裁技术，通过设置优先级来避免冲突。通信距离最远可达 10km（5kbit/s），通信速率最高可达 40 Mbit/s，网络节点数可达 110 个。目前已有多家公司开发了符合 CAN 协议的通信芯片。

3. Lonworks

它由美国 Echelon 公司推出，并由 Motorola、Toshiba 公司共同倡导。它采用 ISO/OSI 模型的全部 7 层通信协议，采用面向对象的设计方法，通过网络变量把网络通信设计简化为参数设置。支持双绞线、同轴电缆、光缆和红外线等多种通信介质，通信速率从 300bit/s～1.5 Mbit/s，直接通信距离可达 2700 m（78 kbit/s），被称为通用控制网络。Lonworks 技术采用的 LonTalk 协议被封装到 Neuron（神经元）的芯片中，并得以实现。采用 Lonworks 技术和神经元芯片的产品，被广泛应用于楼宇自动化、家庭自动化、保安系统、办公设备、交通运输、工业过程控制等领域。

4. DeviceNet

DeviceNet 既是一种低成本的通信连接，也是一种简单的网络解决方案，有着开放的网络标准。DeviceNet 具有的直接互联性不仅改善了设备间的通信，而且提供了相当重要的设备级诊断功能。DeviceNet 基于 CAN 技术，传输速率为 125～500 kbit/s，每个网络的最大节点为 64 个，其通信模式为生产者/客户（Producer/Consumer），采用多信道广播消息发送方式。位于 DeviceNet 网络上的设备可以自由连接或断开，不影响网上的其他设备，而且其设备的安装布线成本也较低。DeviceNet 总线的组织结构是开放式设备网络供应商协会。

5. HART

HART 是 Highway Addressable Remote Transducer 的缩写，最早由 Rosemount 公司开发。其特点是在现有模拟信号传输线上实现数字信号通信，属于模拟系统向数字系统转变的过渡产品。其通信模型采用物理层、数据链路层和应用层三层，支持点对点主从应答方式和多点广播方式。由于它采用模拟和数字信号混合，难以开发通用的通信接口芯片。HART 能利用总线供电，可满足本质安全防爆的要求，并可用于由手持编程器与管理系统主机作为主设备的双主设备系统。

6. CC－Link

CC－Link 是 Control&Communication Link（控制与通信链路系统）的缩写，在 1996 年 11 月，由以三菱电动机公司为主导的多家公司推出，其增长势头迅猛，在亚洲占有较大份额。在其系统中，可以将控制和消息数据同时以 10 Mbit/s 高速传送至现场网络，具有性能卓越、使用简单、应用广泛、节省成本等优点。其不仅解决了工业现场配线复杂的问题，同时具有优异的抗噪性和兼容性。CC－Link 是一个以设备层为主的网络，同时也可覆盖较高层次的控制层和较低层次的传感层。2005 年 7 月，CC－Link 被中国国家标准化管理委员会批准为中国国家标准指导性技术文件。

7. INTERBUS

INTERBUS 是德国 Phoenix 公司推出的较早的现场总线，2000 年 2 月成为国际标准 IEC61158。INTERBUS 采用国际标准化组织（ISO）的开放化系统互连（OSI）的简化模型（1 层、2 层和 7 层），即物理层、数据链路层和应用层。具有强大的可靠性、可诊断性和易维护性。其采用集总帧型的数据环通信，具有低速度、高效率的特点，并严格保证了数据传输的同步性和周期性。该总线的实时性、抗干扰性和可维护性也非常出色。INTERBUS 广泛地应用到汽车、烟草、仓储、造纸、包装、食品等工业领域，成为国际现场总线的领先者。

6.5.3 现场总线的特点

现场总线系统打破了传统控制系统采用的按控制回路要求，设备一对一分别进行连线的结构形式。把原先 DCS 中处于控制室的控制模块、各输入输出模块放入现场设备，加上现场设备具有通信能力，因而控制系统功能能够不依赖控制室中的计算机或控制仪表，直接在现场完成，实现了彻底的分散控制。

现场总线控制系统既是一个开放通信网络，又是一种全分布控制系统。它把作为网络节点的智能设备连接成自动化网络系统，实现基础控制、补偿计算、参数修改、报警、显示、监控、优化的综合自动化功能，是一项以智能传感器、控制、计算机、数字通信、网络为主要内容的综合技术。

现场总线系统具有以下特点：

① 系统具有开放性和互用性。通信协议遵从相同的标准，设备之间可以实现消息交换，可按需要把不同供应商的产品组成开放互连的系统。系统间、设备间可以进行消息交换，不同生产厂家的性能类似的设备可以互换。

② 系统功能自治性。系统将传感测量、补偿计算、工程量处理与控制等功能分散到现场设备中完成，现场设备可以完成自动控制的基本功能，并可以随时诊断设备的运行状况。

③ 系统具有分散性。现场总线构成的是一种全分散的控制系统结构，简化了系统结构，提高了可靠性。

④ 系统具有对环境的适应性。现场总线支持双绞线、同轴电缆、光缆、射频、红外线、电力线等，具有较强的抗干扰能力，能采用两线制实现供电和通信，并可以满足安全防爆的要求。

6.5.4 现场总线的现状

1. 多种现场总线并存

目前世界上存在着40余种现场总线，如法国的FIP、英国的ERA、德国西门子公司的PROFIBUS、挪威的FINT、Echelon公司的LONWorks、Phoenix Contact公司的INTERBUS、Rober Bosch公司的CAN、Rosemount公司的HART、Carlo Garazzi公司的Dupline、丹麦Process Data公司的P－net、Peter Hans公司的F－Mux、ASI（Actratur Sensor Interface）、MODBus、SDS、Arcnet、国际标准组织－基金会现场总线FF（Field Bus Foundation，World-FIP）、BitBus、美国的DeviceNet与ControlNet等。这些现场总线用于过程自动化、医药、加工制造、交通运输、国防、航天、农业和楼宇等领域，不到10种类型的总线占有80%左右的市场。

2. 各种总线都有其应用的领域

每种总线都有其应用的领域，如FF和PROFIBUS－PA适用于石油、化工、医药、冶金等行业的过程控制领域；LonWorks、PROFIBUS－FMS和DevieceNet适用于楼宇、交通运输、农业等领域；DeviceNet、PROFIBUS－DP适用于加工制造业。这些划分也不是绝对的，每种现场总线都力图将其应用领域扩大，彼此渗透。

3. 每种现场总线都有其国际组织和支持背景

大多数的现场总线都有一个或几个大型跨国公司为背景并成立相应的国际组织，力图扩大自己的影响，得到更多的市场份额，如PROFIBUS以Siemens公司为主要支持，并成立了PROFIBUS国际用户组织；World FIP以Alstom公司为主要支持，成立了World FIP国际用户组织。

4. 多种总线成为国家和地区标准

为了加强自己的竞争能力，很多总线都争取成为国家或者地区的标准，如PROFIBUS已成为德国标准，World FIP已成为法国标准等。

5. 设备制造商参与多个总线组织

为了扩大自己产品的使用范围，很多设备制造商往往参与多个总线组织。

6. 各个总线彼此协调共存

由于竞争激烈，而且还没有哪一种或几种总线能一统市场，很多重要企业都力图开发接口技术，使自己的总线能和其他总线相连，在国际标准中也出现了协调共存的局面。

6.5.5 现场总线的发展

现场总线技术是控制、计算机和通信技术的交叉与集成，几乎涵盖了连续和离散工业领域，如过程自动化、制造加工自动化、楼宇自动化、家庭自动化等。

它的出现和快速发展体现了控制领域对降低成本、提高可靠性、增强可维护性和提高数

据采集智能化的要求。现场总线技术的发展趋势体现在四个方面。

1. 统一的技术规范与组态技术是现场总线技术发展的一个长远目标

IEC61158 是目前的国际标准。然而由于商业利益的问题，该标准只做到了对已有现场总线的确认，从而得到了各个大公司的欢迎。但是却给用户带来了使用的困难。当需要用一种新的总线的时候，学习的过程是漫长的。从长远来看，各种总线的统一是必由之路。目前主流现场总线都是基于 EIA485 技术或以太网技术，有了统一的硬件基础。组态的过程与操作是相似的，有了统一的用户基础。

2. 现场总线系统的技术水平将不断提高

随着电子技术、网络技术和自动控制技术的发展，现场总线设备将具备更强的性能、更高的可靠性和更好的经济性。

3. 现场总线的应用将越来越广泛

随着现场总线技术的日渐成熟，相关产品的性价比越来越高，更多的技术人员将掌握现场总线的使用方法，现场总线的应用将越来越广泛。

4. 工业以太网技术将逐步成为现场总线技术的主流

虽然基于串行通信的现场总线技术在一段时期之内还会大量使用，但是从发展的眼光来看，工业以太网具有良好的适应性、兼容性、扩展性以及与信息网络的无缝连接等特性，必将成为现场总线技术的主流。

SIMATIC NET 是西门子工业网络通信解决方案的统称。

6.5.6 S7 - 200 PLC PROFIBUS 通信的应用

PROFIBUS - DP 是一种通信标准，支持 PROFIBUS - DP 协议的第三方设备都会有 GSD 文件，通常以 *. GSD 或者 *. GSE 文件出现，将此文件安装到 STEP 7 软件中，才能组态第三方设备从站的通信接口。例如正常安装的 STEP 7 软件中是不能组态第三方设备 EM 277 的，必须安装“siem089d. gsd”文件才能组态 EM 277。

【关键点】 没有 S7 - 300/400 PLC 参与，S7 - 200 PLC 间不能进行 PROFIBUS 通信。

下面以一台 CPU 314C - 2DP 与一台 CPU 226CN 之间的 PROFIBUS 的现场总线通信为例介绍 S7 - 300 系列 PLC 与第三方设备的 PROFIBUS - DP 通信。

【例 6-7】 模块化生产线的主站为 CPU 314C - 2DP，从站为 CPU 226CN 和 EM 277 的组合，主站发出开始信号（开始信号为高电平），从站接收信息，并使从站的指示灯以 1 s 为周期闪烁。

【解】

1. 主要软硬件配置

① 1 套 STEP 7 - Micro/WIN V4. 0 SP9。

② 1 套 STEP 7 V5. 5 CN。

③ 1 台 CPU 226CN。

④ 1 台 EM 277。

⑤ 1 台 CPU 314C - 2DP。

⑥ 1 根 PC/PPI 电缆和 1 根 PC/MPI 适配器（或者 CP 5611 卡）。

⑦ 1 根 PROFIBUS 电缆（含两个网络总线连接器）。

PROFIBUS 现场总线硬件配置图如图 6-53 所示，PROFIBUS 现场总线通信 PLC 接线图如图 6-54 所示。

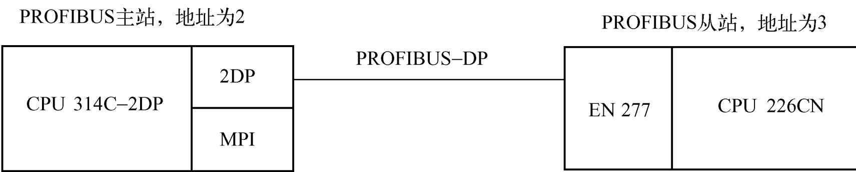


图 6-53　PROFIBUS 现场总线硬件配置图

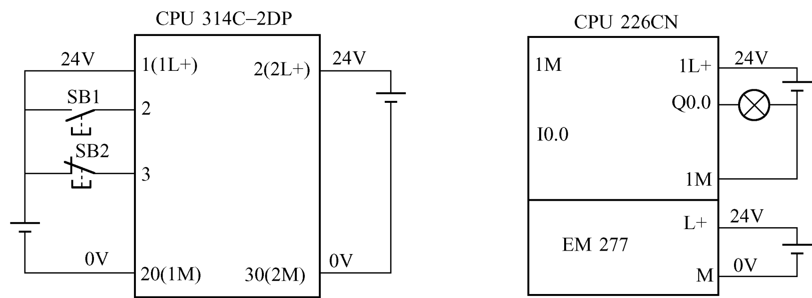


图 6-54　PROFIBUS 现场总线通信 PLC 接线图

2. CPU 314C－2DP 的硬件组态

S7－300 PLC 与 S7－200 PLC 的 PROFIBUS 通信总的方法是：首先对主站 CPU 314C－2DP 的硬件进行硬件组态，下载硬件，再编写主站程序，下载主站程序。编写从站程序，下载从站程序，最后便可建立主站和从站的通信。具体步骤如下：

1）打开 STEP 7 软件。双击桌面上的快捷键“”，打开 STEP 7 软件，如图 6-55 所示。当然也可以单击“开始”→“所有程序”→“SIMATIC”→“SIMATIC Manager”，打开 STEP 7 软件。



图 6-55　打开 STEP 7 软件

2）新建项目。单击“新建”按钮“🗋”，弹出“新建 项目”对话框，在“命名（M）”中输入一个名称，本例为“profibus”，之后再单击“确定”按钮，如图 6-56 和图 6-57 所示。

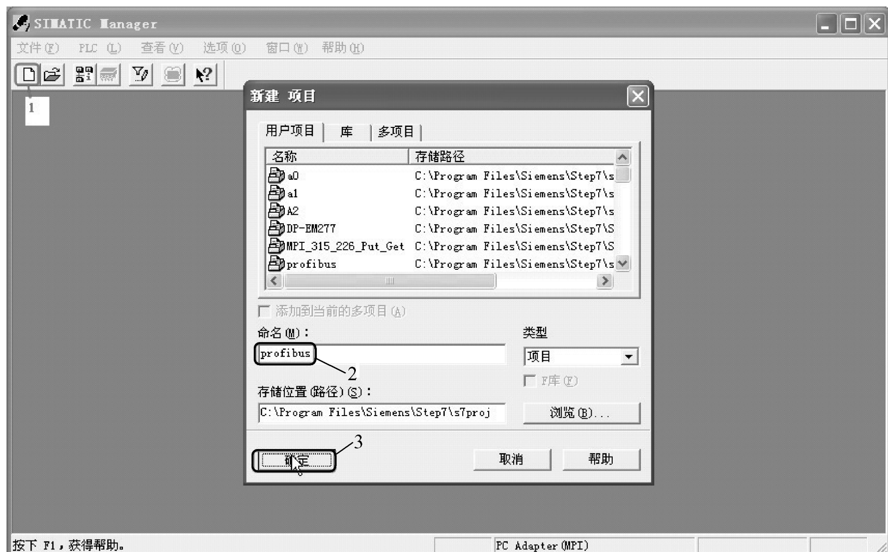


图 6-56 新建项目（1）

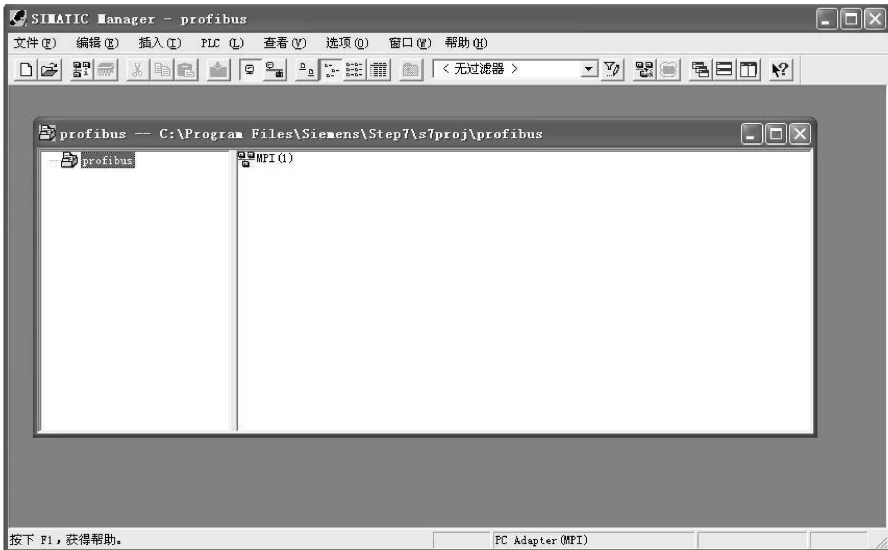


图 6-57 新建项目（2）

3）插入站点。单击菜单栏“插入”菜单，再单击“站点”和“SIMATIC 300 站点”子菜单，如图 6-58 和图 6-59 所示。这个步骤的目的主要是为了插入主站。

4）插入导轨。展开项目中的“SIMATIC 300”下的“RACK-300”，双击导轨“Rail”，如图 6-60 所示。硬件配置的第一步都是加入导轨，否则下面的步骤不能进行。



图 6-58　插入站点（1）

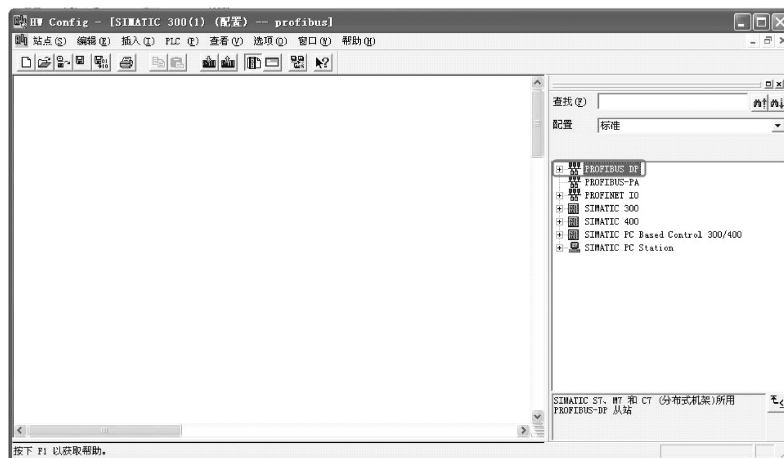


图 6-59　插入站点（2）

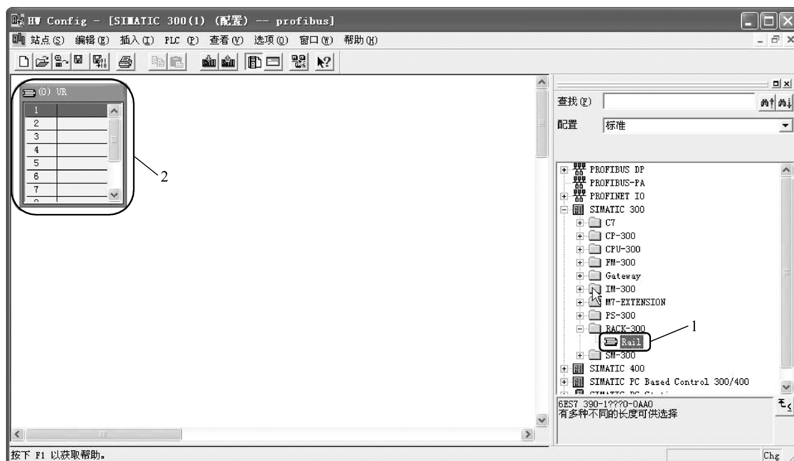


图 6-60　插入导轨

5）插入 CPU。展开项目中的“SIMATIC 300”下的“CPU－300”，再展开“CPU 314C－2DP”下的“6ES7 314－6CG03－0AB0”，将“V2.6”拖入导轨的2号槽中，如图6-61所示。若选用了西门子的电源，在配置硬件时，应该将电源加入到第一槽，本例中使用的是开关电源，因此硬件配置时不需要加入电源，但第一槽必须空槽，建议读者最好选用西门子电源。



图6-61　插入 CPU

6）配置网络。双击2号槽中的“DP”，弹出“属性－DP”对话框，单击“属性”按钮，再弹出“属性－PROFIBUS 接口”对话框，如图6-62所示。单击“新建”按钮，弹出“属性－新建子网 PROFIBUS”对话框，如图6-63所示。选定传输速率为“1.5Mbit/s”和配置文件为“DP”，单击“确定”按钮，如图6-64所示。从站便可以挂在 PROFIBUS 总线上。



图6-62　新建网络



图 6-63 设置通信参数

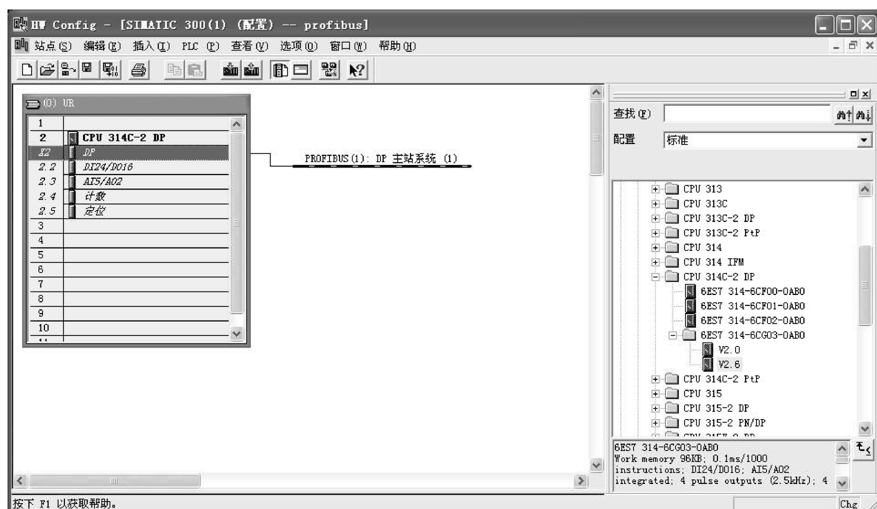


图 6-64 配置网络

7）修改 I/O 起始地址。双击 2 号槽中的“DI24/DO16”，弹出“属性 – DI24/DO16”对话框，如图 6-65 所示。去掉“系统默认”前的“√”，在“输入”和“输出”的“开始”中输入“0”，单击“确定”按钮，如图 6-66 所示。这个步骤目的主要是为了使程序中输入和输出的起始地址都从“0”开始，这样更加符合我们的习惯，若没有这个步骤，也是可行的，但程序中输入和输出的起始地址都从“124”开始，不方便。

8）配置从站地址。先选中“PROFIBUS”，再展开项目，先后展开“PROFIBUS – DP”→“Additional Field Device”→“PLC”→“SIMATIC”，再双击“EM 277 PROFIBUS – DP”，弹出“属性 – PROFIBUS 接口”对话框，将地址改为“3”，最后单击“确定”按钮，如图 6-67 所示。

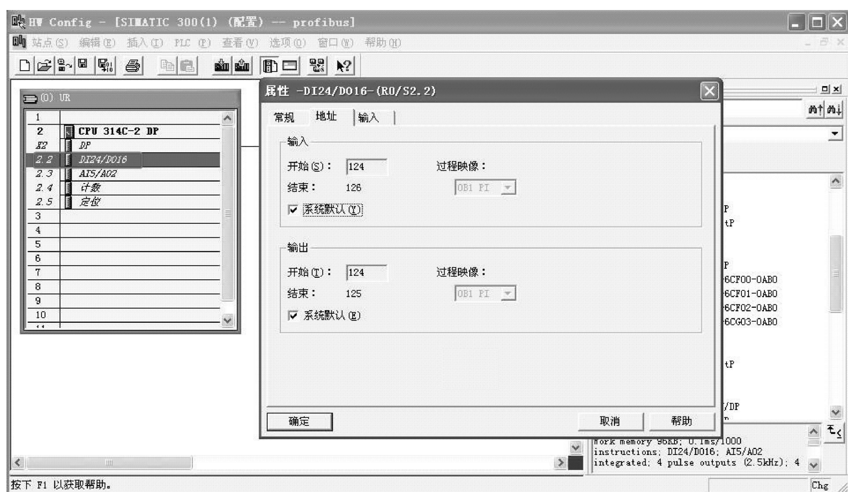


图 6-65　修改 I/O 起始地址（1）

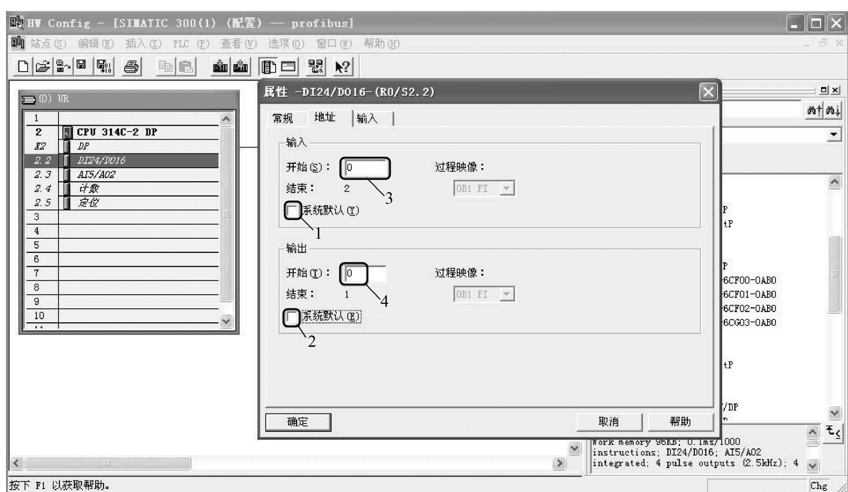


图 6-66　修改 I/O 起始地址（2）



图 6-67　配置从站地址

【关键点】由于 EM 277 是第三方设备，默认状态并不被 STEP 7 软件支持，所以必须安装 GSD 文件“siem089D. gsd”，此文件可在西门子驱动的官方网站上免费下载。

9）分配从站通信数据存储区。先选中 3 号站，再展开项目“EM 277 PROFIBUS - DP”，再双击“1 Word In/1 Word Out”，如图 6-68 所示。当然也可以选其他的选项，这个选项的含义是：每次主站接收信息为 1 个字，送出的信息也为 1 个字。

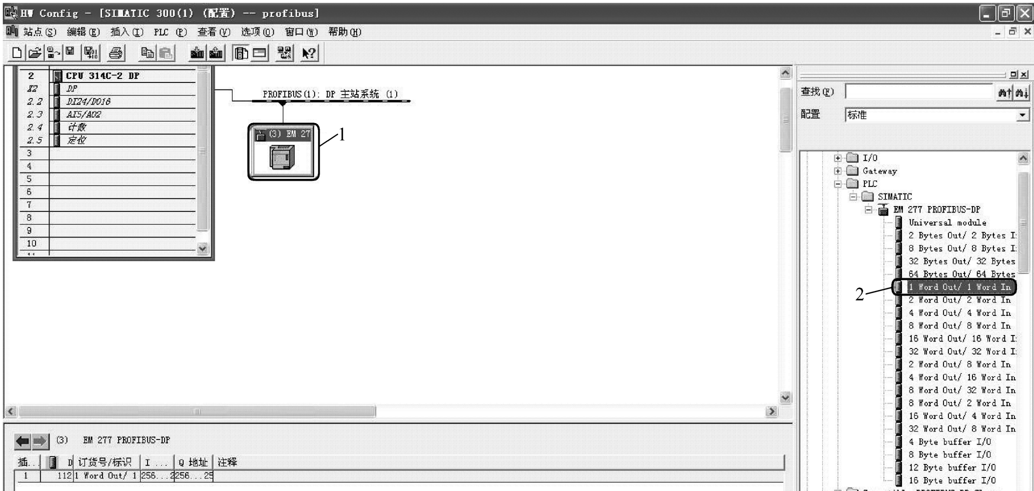


图 6-68　分配从站通信数据存储区

10）修改通信数据发送区和接收数据区起始地址。先选中 3 号站下的数据的接收和发送区，双击，弹出“属性 - DP 从站”对话框，如图 6-69 所示。再在输入的启动地址中输入“3”，输出启动地址中输入“2”，如图 6-70 所示，再单击“确定”按钮。这样做的目的是为了使后续的程序的输入、输出地址更加符合我们的习惯，这个步骤可以没有。

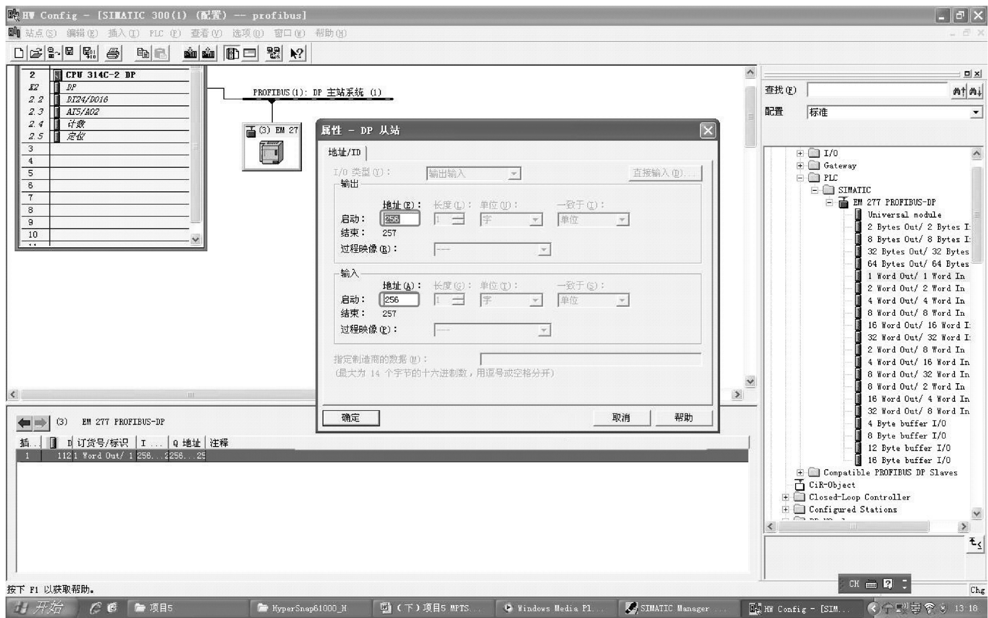


图 6-69　修改通信数据发送区和接收数据区起始地址（1）

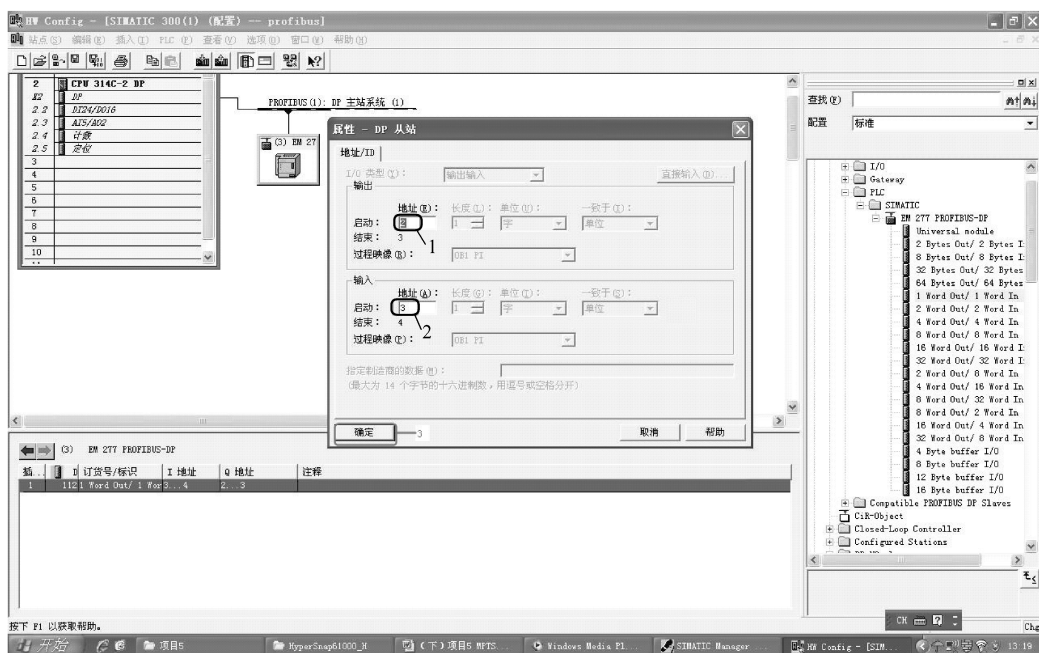


图 6-70　修改通信数据发送区和接收数据区起始地址（2）

11）下载硬件组态。到目前为止，已经完成了硬件的组态，单击“保存和编译”按钮“ ”，若有错误，则会显示，没有错误，系统将自动保存硬件组态。接着单击“下载”按钮“ ”，系统将硬件配置下载到 PLC 中。下载硬件的步骤是不可以缺少的，否则前面所做的硬件配置的工作都是徒劳，但保存和编译步骤可以省略，因为单击下载按钮也可以起到这个作用。

12）打开块并编译程序。激活“SIMATIC Manager - profibus”界面，展开工程“profibus”，选中“块”，如图 6-71 所示。单击“OB1”图标，弹出“属性 - 组织块”对话框，再单击“确定”按钮，如图 6-72 所示。之后弹出“LAD/STL/FBD”界面，实际上是程序编辑界面，在此界面上，输入如图 6-73 所示的程序。

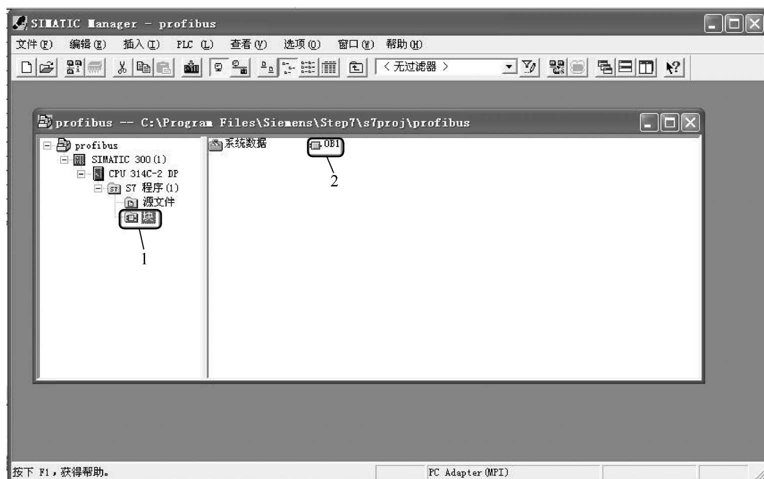


图 6-71　打开块并编译程序（1）



图 6-72　打开块并编译程序（2）

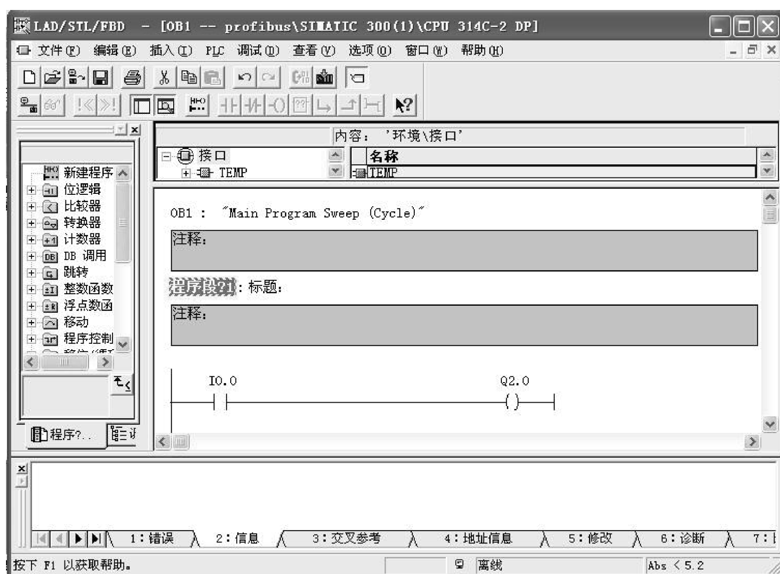


图 6-73　CPU 314C－2DP 的程序

3. 编写程序

1）编写主站的程序。按照以上步骤进行硬件组态后，主站和从站的通信数据发送区和接收数据区就可以进行数据通信了，主站和从站的发送区和接收数据区对应关系见表 6-18。

表 6-18　主站和从站的发送区和接收数据区对应关系

序　　号	主站 S7－300	对 应 关 系	S7－200 从站
1	QW2	⟶	VW0
2	IW3	⟵	VW2

主站将信息存入 QW2 中，发送到从站的 VW0 数据存储区，那么主站的发送数据区为什么是 QW2 呢？因为 CPU 314C－2DP 自身是 16 点数字输出占用了 QW0，因此不可能是 QW0，QW2 是在前面的序（10）中设定的。当然也可以设定为其他的单元，但不可以设定为 QW0。从站的接收区默认为 VW0，从站的发送区默认为 VW2，这个单元是可以在硬件组态时更改的，请读者参考西门子的相关手册。从站的信息可以通过 VW2 送到主站的 IW3。注意，务必要将组态后的硬件和编译后的软件全部下载到 PLC 中。

2）编写从站程序。在桌面上双击快捷键“”，打开软件 STEP 7－Micro/WIN，在梯形图中输入如图 6-74 的程序。再将程序下载到从站 PLC 中。图 6-74 中的程序的含义是：当从站收到信号时，VW0 大于 1，M10.0 自锁输出，Q0.0 以 1 s 的频率闪烁。

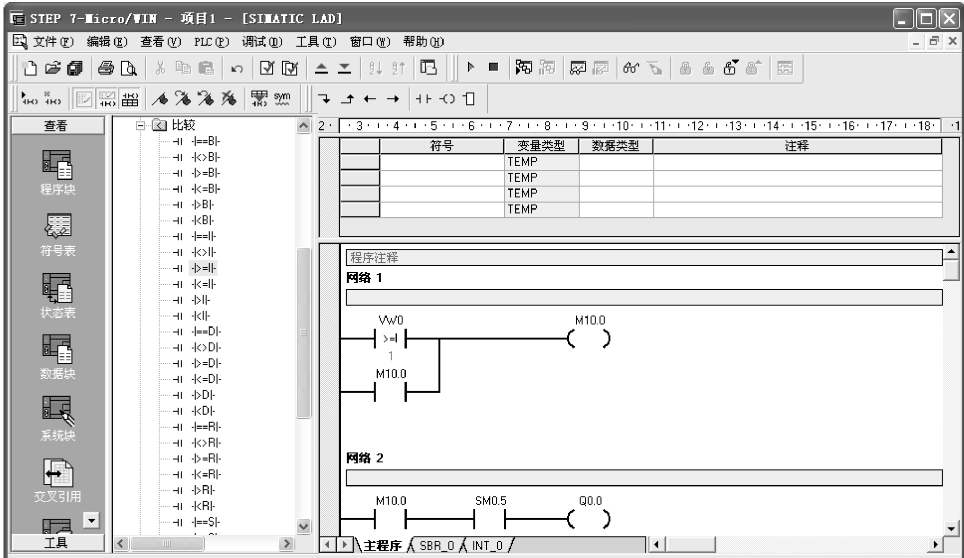


图 6-74　CPU 226CN 的程序

4. 硬件连接

主站 CPU 314C－2DP 有两个 DB9 接口：一个是 MPI 接口，它通过 S7－300 专用编程电缆与计算机相连（也可作为 MPI 通信使用）；另一个 DB9 接口是 DP 口，PROFIBUS 通信使用这个接口。从站为 CPU 226CN 和 EM 277，EM 277 是 PROFIBUS 专用模块，这个模块上面 DB9 接口为 DP 口。主站的 DP 口和从站的 DP 口使用专用的 PROFIBUS 电缆和专用网络接头相连。

PROFIBUS 电缆是二线屏蔽双绞线，两根线为 A 线和 B 线，电线塑料皮上印刷有 A、B 字母，A 线与网络接头上的 A 端子相连，B 线与网络接头上的 B 端子相连即可。B 线实际与 DB9 型接插件的第 3 针相连，A 线实际与 DB9 型接插件的第 8 针相连。

【关键点】在前述的硬件组态中已经设定从站为第 3 站，因此在通信前，必须要将 EM277 的“站号”选择旋钮旋转到“3”的位置，否则，通信不可能成功。此外，完成设定 EM277 的站地址后，必须将 EM277 断电，新设定的站地址才能生效。从站网络连接器的终端电阻应置于“on”，如图 6-75 所示。若要置于“on”，只要将拨钮拨向“on”一侧即可。

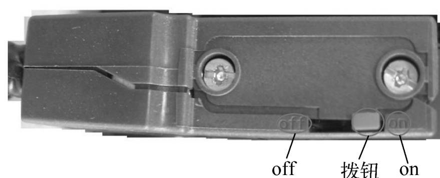


图 6-75　网络连接器的终端电阻置于“on”

5. 软硬件调试

用 PROFIBUS 的电缆将 S7－300 的 DP 口与 EM 277 的 DP 口相连，并将 S7－300 端的网络连接器上的拨钮拨到“off”，并将 EM 277 端的网络连接器上的拨钮拨到“on”上。再将程序下载到 PLC 中。最后将两台 PLC 的运行状态从“STOP”都拨到“RUN”上。

6.6　以太网通信

6.6.1　工业以太网通信简介

1. 初识工业以太网

所谓工业以太网，通俗地讲就是应用于工业的以太网，是指其在技术上与商用以太网（IEEE802.3 标准）兼容，但材质的选用、产品的强度和适用性方面应能满足工业现场的需要。工业以太网技术的优点表现在：以太网技术应用广泛，为所有的编程语言所支持；软硬件资源丰富；易于与 Internet 连接，实现办公自动化网络与工业控制网络的无缝连接；通信速度快；可持续发展的空间大等。

虽然以太网有众多的优点，但作为信息技术基础的 Ethernet 是为 IT 领域应用而开发的，在工业自动化领域只得到有限应用，这是由于以下原因：

① Ethernet 采用 CSMA/CD 碰撞检测方式，在网络负荷较重时，网络的确定性（Determinism）不能满足工业控制的实时要求。

② Ethernet 所用的接插件、集线器、交换机和电缆等是为办公室应用而设计，不符合工业现场恶劣环境要求。

③ 在工程环境中，Ethernet 抗干扰（EMI）性能较差。若用于危险场合，以太网不具备本质安全性能。

④ Ethernet 网还不具备通过信号线向现场仪表供电的性能。

随着信息网络技术的发展，上述问题正在迅速得到解决。为促进 Ethernet 在工业领域的应用，国际上成立了工业以太网协会（Industrial Ethernet Association，IEA）。

2. 网络电缆接法

用于 Ethernet 网的双绞线有 8 芯和 4 芯两种，双绞线的电缆连线方式也有两种，即正线（标准 568B）和反线（标准 568A），其中正线也称为直通线，反线也称为交叉线。正线接线如图 6-76 所示，两端线序一样，从上至下线序是：白绿，绿，白橙，蓝，白蓝，橙，白棕，棕。反线接线如图 6-77 所示，一端为正线的线序，另一端为从上至下线序是：白橙，橙，白绿，白棕，棕，绿，蓝，白蓝。对于千兆以太网，用 8 芯双绞线，但接法不同以上所述的接法，请参考有关文献。

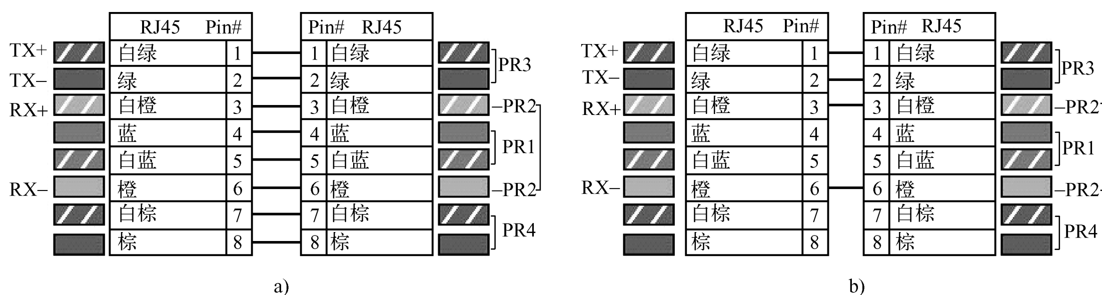


图 6-76　双绞线正线接线图
a）8 芯线　b）4 芯线

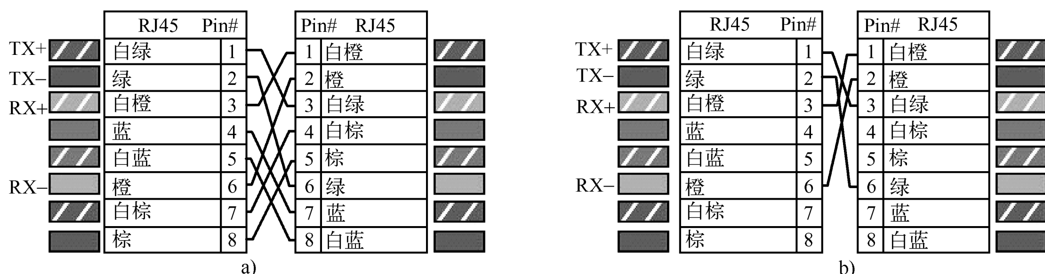


图 6-77　双绞线反线接线图
a）8 芯线　b）4 芯线

对于 4 芯的双绞线，只用连接头上的（常称为水晶接头）1、2、3 和 6 四个引脚。西门子的 PROFINET 工业以太网采用 4 芯的双绞线。

常见的采用正线连接的有：计算机（PC）与集线器（HUB）、计算机（PC）与交换机（SWITCH）、PLC 与交换机（SWITCH）、PLC 与集线器（HUB）。

常见的采用反线连接的有：计算机（PC）与计算机（PC）、PLC 与 PLC。

6.6.2　S7 -200 PLC 以太网通信的应用

S7 -200 系列 PLC 自身不带以太网接口，因此要组成以太网必须配备以太网模块 CP243 -1 或者 CP243 -1 IT。

S7 -200 PLC 间的以太网通信只能用 S7 方式进行，其中一台作为客户端（Client），其余的为服务器端（Server）。以下用一个例子介绍两台 S7 -200 间的以太网通信。

【例 6-8】一台 S7 -200 作为服务器端，另一台 S7 -200 作为客户端，当服务器端发出一个起停信号时，客户端收到信号并起停一台电动机。

【解】

1. 软硬件配置

① 2 台 CPU 226CN。

② 2 台 CP 243 -1 IT 以太网模块。

③ 1 台 8 口交换机。

④ 1 根 PC/PPI 电缆（USB 口）。

⑤ 2 根带水晶接头的 8 芯双绞线（正线或者反线）。
⑥ 1 台个人电脑（含网卡）。
⑦ 1 套 STEP 7 – Micro/WIN V4.0 SP9。
S7 – 200 间的以太网通信硬件配置图如图 6-78 所示，接线图如图 6-79 所示。

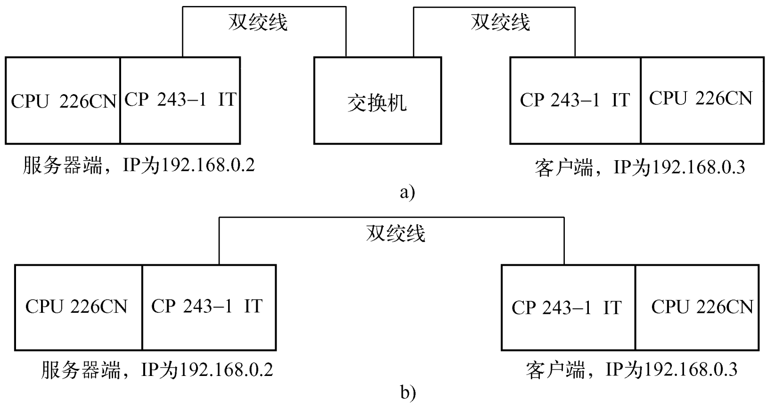


图 6-78　S7 – 200 间的以太网通信硬件配置图
a）方案 1　b）方案 2

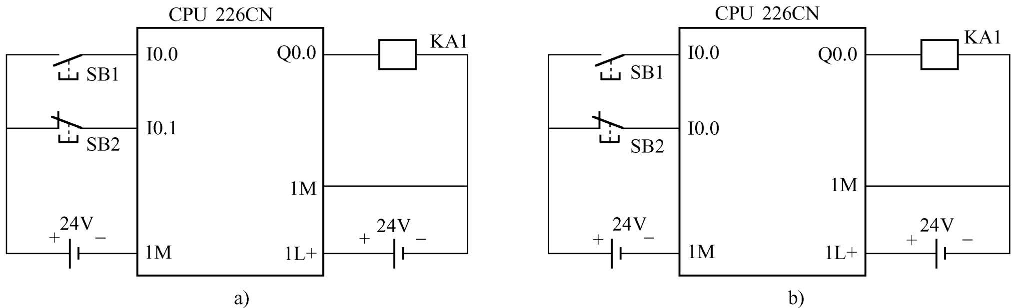


图 6-79　接线图

方案 1 中的网线可以是正连接和反连接，原因在于交换机具有自动交叉线功能，而方案 2 的网线只能是正连接。

【关键点】 方案 1 使用的双绞线是反线连接，由于交换机具有自动交叉网线（auto – cross – over）功能，所以采用正线连接也是可行的，当然交换机换成集线器（HUB）也可以，但最佳的选择就是使用工业交换机。方案 2（不经过交换机或者 HUB），那么只能采用正线连接，而不能采用反线连接。

2. 将一台 S7 – 200 设置成服务器端

要实现 S7 – 200 PLC 间的以太网通信，首先要对 S7 – 200 进行设置，并利用 STEP 7 – Micro/WIN 的“以太网向导”生成通信，以下为具体的步骤：

1）首先建立 STEP 7 – Micro/WIN 与 S7 – 200 之间的通信。启动 STEP 7 – Micro/WIN 编程软件，新建项目并命名为“EtanServer”，读取或者设置 PLC 的类型，再单击“确定”按钮，如图 6-80 所示。

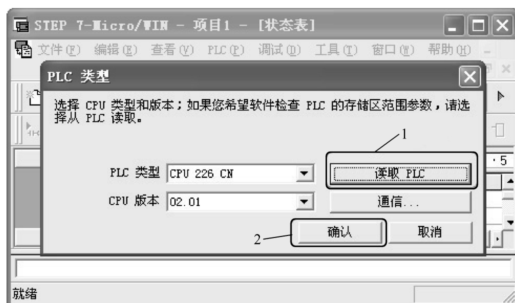


图 6-80　读取 PLC 的类型

在“查看”中，单击“通信”按钮，弹出“通信”界面，再单击“设置 PG/PC 接口”按钮，如图 6-81 所示。



图 6-81　设置 PG/PC 接口

如图 6-82 所示，先选中“PC/PPI（PPI）”选项，再单击“属性”按钮，弹出如图 6-83属性界面，选中“USB”接口（若使用的是 RS-232C 接口的 PC/PPI，则此处应选择 COM1），单击“确定”按钮，回到图 6-82 界面，再单击“确定”按钮。

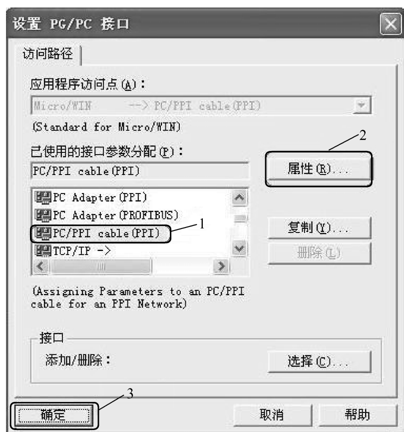


图 6-82　设置通信参数

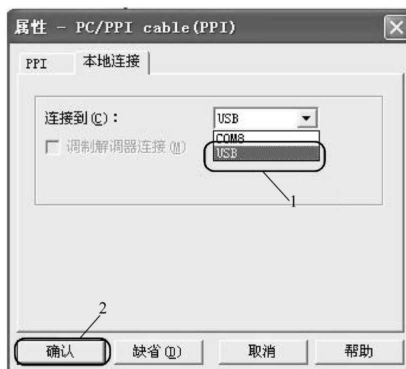


图 6-83　设置计算机的接口

此时回到如图 6-81 的界面，双击“双击刷新”，找到“CPU 226 CN”，其地址为 2，如图 6-84 所示，单击“确定”按钮。



图 6-84　设置计算机的接口

2）打开“以太网向导”。单击“工具”→“以太网向导”，弹出“以太网向导”初始界面，如图 6-85 和图 6-86 所示，单击“下一步”按钮。



图 6-85　打开以太网向导

3）指定模块位置。在模块位置中输入位置号，本例为“0”，再单击“读取模块”按钮，若读取成功，则模块的信息显示在如图 6-87 所示的序号“3”处，单击“下一步”按钮。

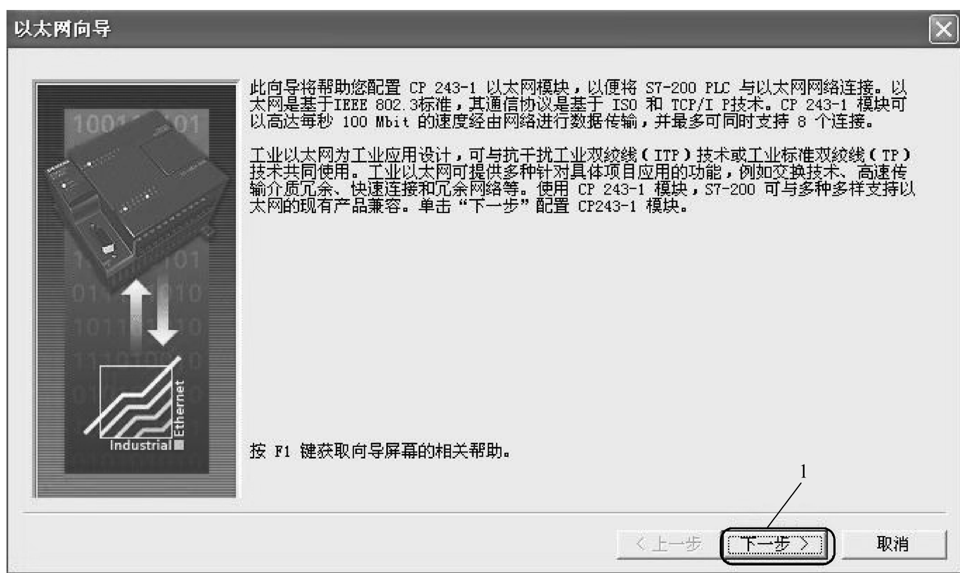


图 6-86　以太网向导初始界面

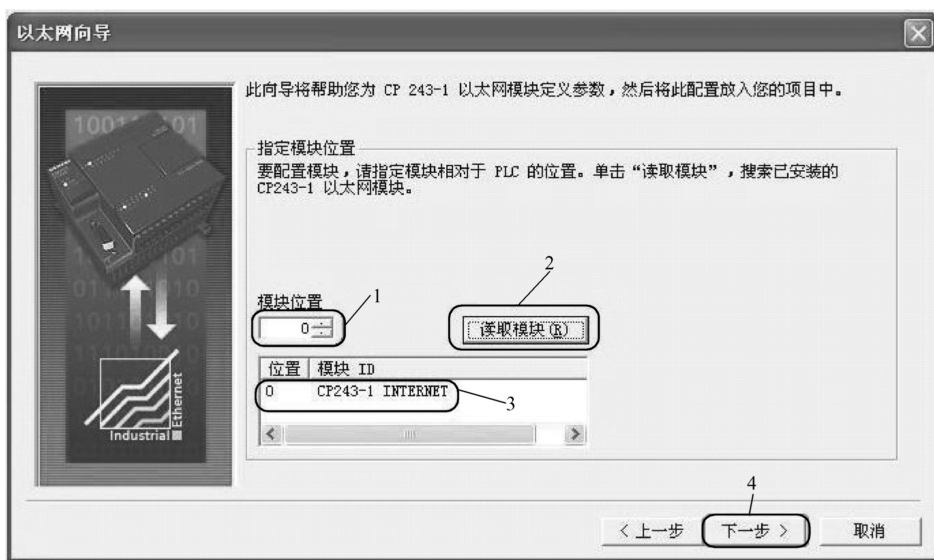


图 6-87　指定模块位置

4）指定模块地址。在 IP 地址中输入“192. 168. 0. 2”（也可以是其他有效 IP 地址），在“子网掩码”中输入“255. 255. 255. 0”，网关可以空置，“为此模块指定通信连接类型”中选择“自动检测通信”选项，最后单击“下一步”按钮，如图 6-88 所示。

5）指定命令字节和连接数目。因为只有两个模块，所以“为此模块配置的连接数”选定为“1”，再单击“下一步”按钮，如图 6-89 所示。

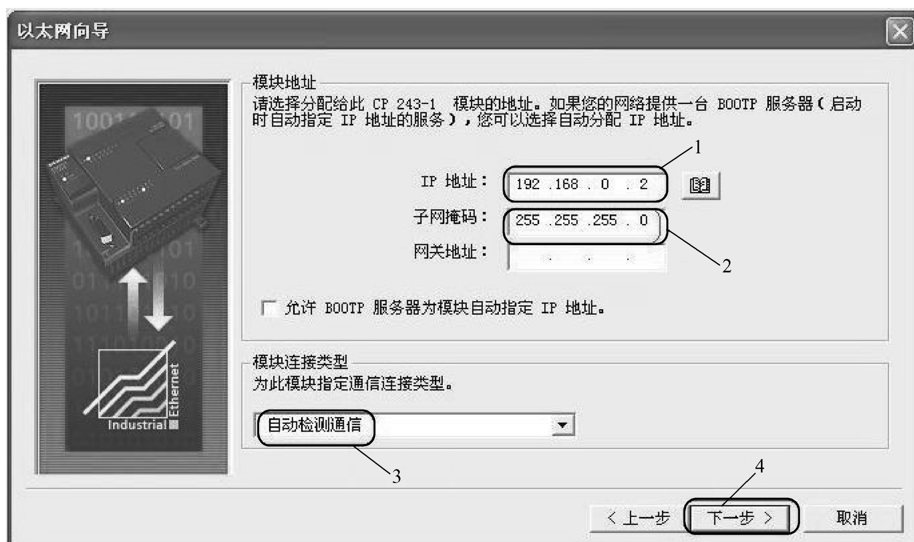


图 6-88　指定模块地址

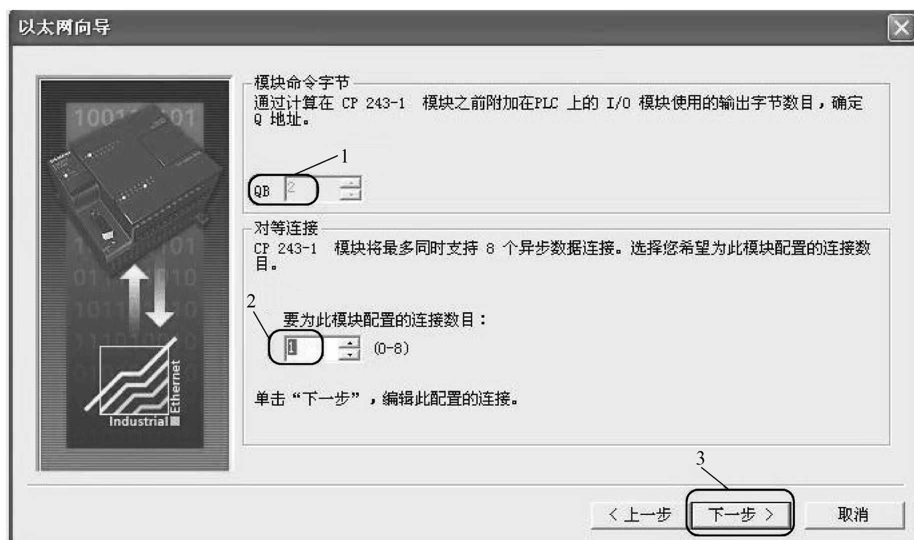


图 6-89　指定命令字节和连接数目

6）配置连接。将客户端的“远程属性”设置为“10.00”，再将客户端的 IP 地址设定“192.168.0.3”，注意，客户端 IP 地址必须与服务器端的 IP 地址在一个网段，否则不能通信。再勾选序号“3”处，单击“确定”按钮，如图 6-90 所示。

7）CRC 保护与保持现用间隔。选中序号“1”处的选项，单击“下一步”按钮，如图 6-91 所示。

8）分配配置内存。单击“建议地址”按钮，再单击“下一步”按钮，如图 6-92 所示。注意生成的建议地址 VB159 ~ VB317 供通信使用，读者编程不可使用。

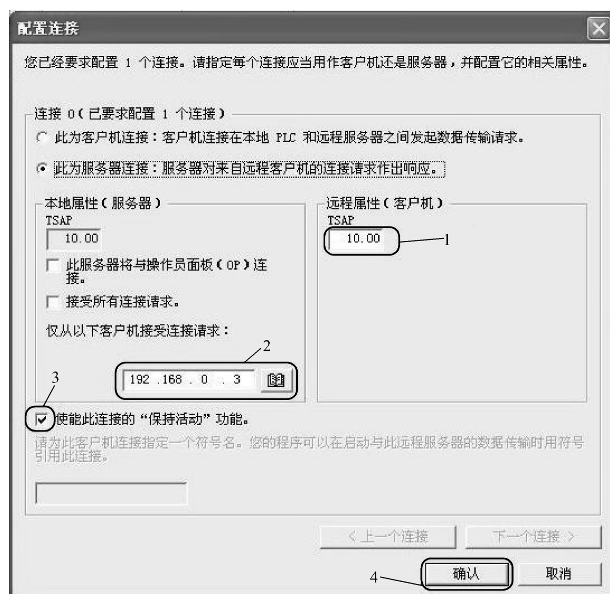


图 6-90　配置连接

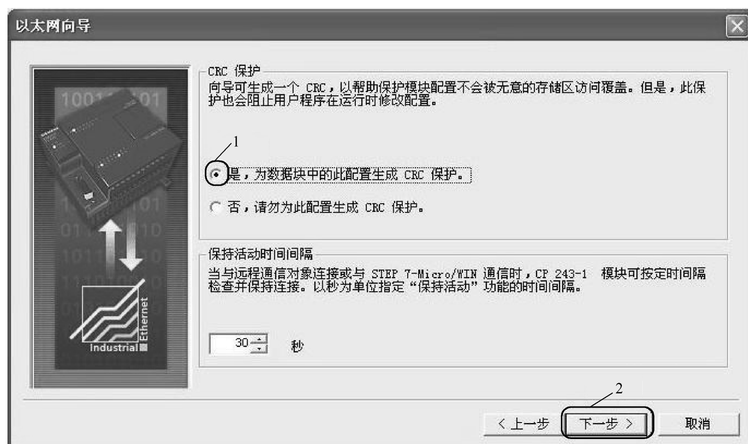


图 6-91　CRC 保护与保持现用间隔



图 6-92　分配配置内存

9）生成项目部件。单击“完成”按钮，以太网向导完成，如图 6-93 所示，STEP 7 - Micro/WIN 自动生成通信指令。

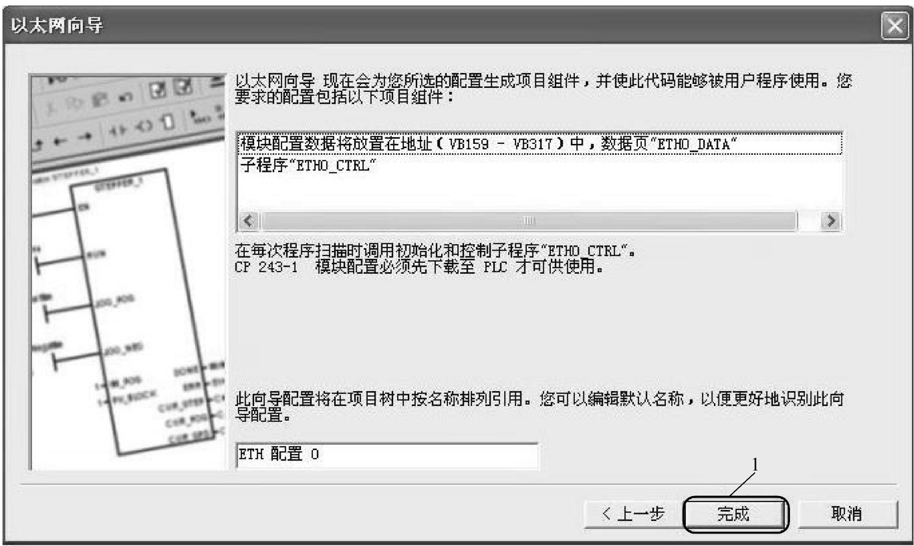


图 6-93　生成项目部件

3. 将另一台 S7 -200 设置成客户端

将 S7 -200 设置成客户端的步骤 1）~3）与将 S7 -200 设置成服务器端的步骤相同，在此不再赘述，只是在步骤 1）中将工程命名为“EtanClient”，以下将从步骤 4）开始。

4）指定模块地址。在 IP 地址中输入“192. 168. 0. 3”，在“子网掩码”中输入“255. 255. 255. 0”，网关可以空置，“为此模块指定通信连接类型”中选择“自动检测通信”选项，最后单击“下一步”按钮，如图 6-94 所示。IP 地址的末位可以为小于等于 255，但应为除 2 外的所有整数。

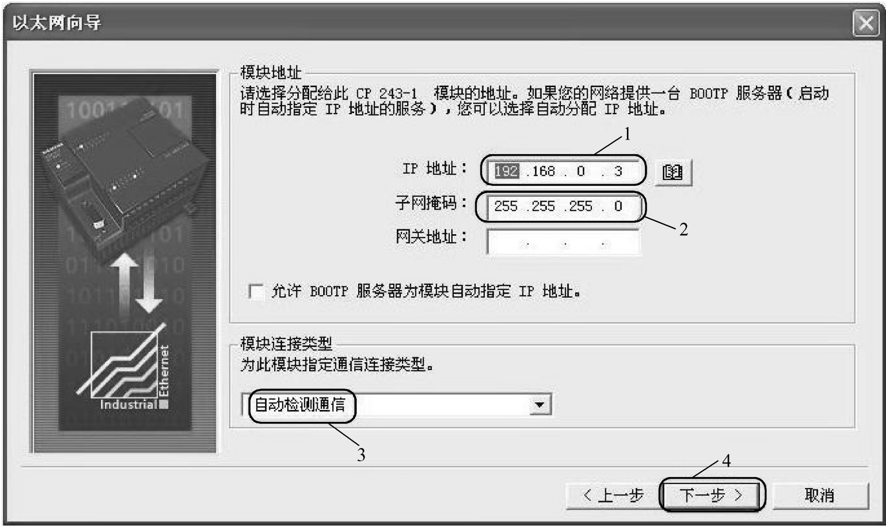


图 6-94　指定模块地址

5）指定命令字节和连接数目。因为只有两个模块，所以“为此模块配置的连接数”选定为“1”，再单击“下一步”按钮，如图 6-95 所示。

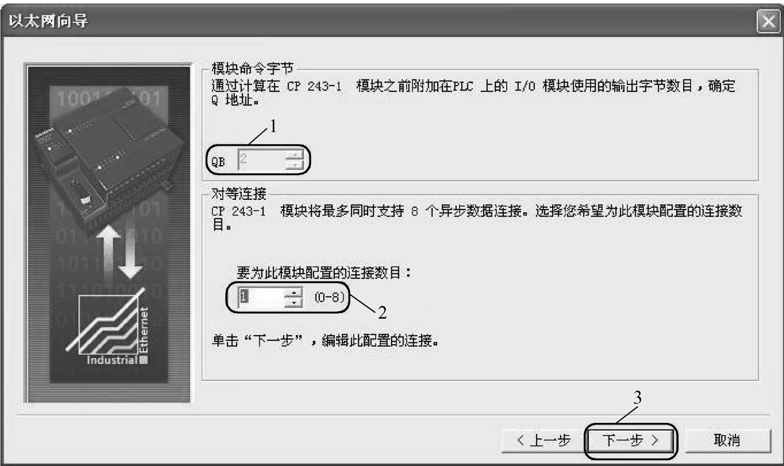


图 6-95　指定命令字节和连接数目

6）配置连接。将“远程属性”设置为“10.00”，再将“为此连接服务器端的 IP 地址”设定“192.168.0.3”，注意，服务器端的 IP 地址必须是前面设置服务器端的 IP 地址，否则不能通信。单击“确定”按钮，如图 6-96 所示。

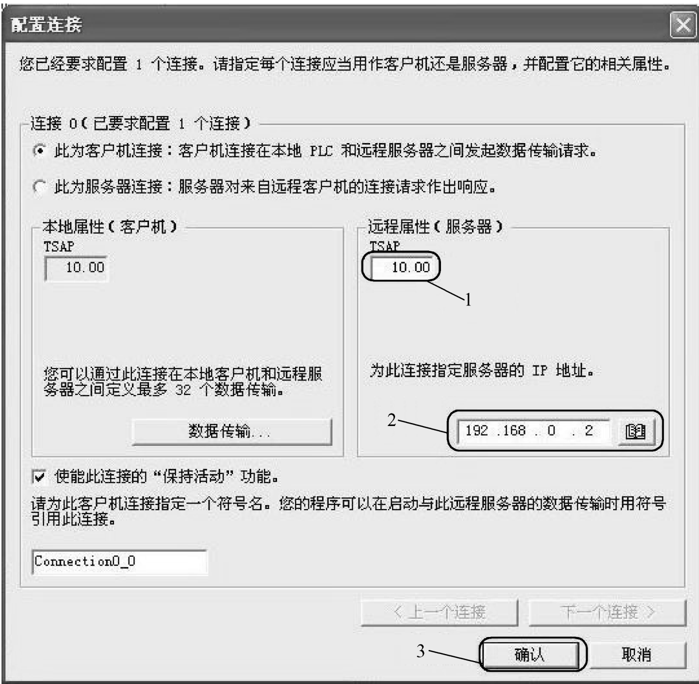


图 6-96　配置连接（1）

有关远程对象的 TSAP 的含义如下：

1）如果连接的远程对象是 S7－200 PLC，使用以下算法确定远程 TSAP。

① TSAP 的第一个字节是 0x10 + 连接数目。

② TSAP 的第二个字节是模块位置。

2）如果连接远程对象是 S7 - 300 或 S7 - 400，使用以下算法确定远程 TSAP。

① TSAP 的第一个字节是 0x3 + 连接数目。

② TSAP 的第二个字节代表模块架和槽位的编码数值。字节的第三个位是模块架，最后 5 个位是编码槽号。

单击“新传输”按钮，再单击“确定”按钮，如图 6-97 所示。

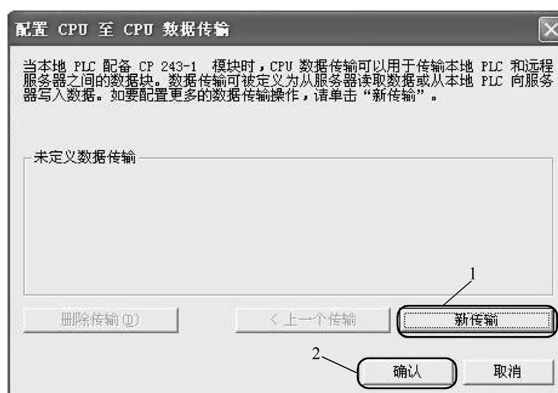


图 6-97　配置连接（2）

如图 6-98 所示，在“1”处，选定“从远程服务器端读取数据”，“2”处选定 1，因为一个字节可以包含 8 个开关量信息，而本例只有一个开关量。“3”和“4”处的含义是将服务器端的 VB0 中的数据传送到客户端的 VB0 中去。再单击“确定”按钮。

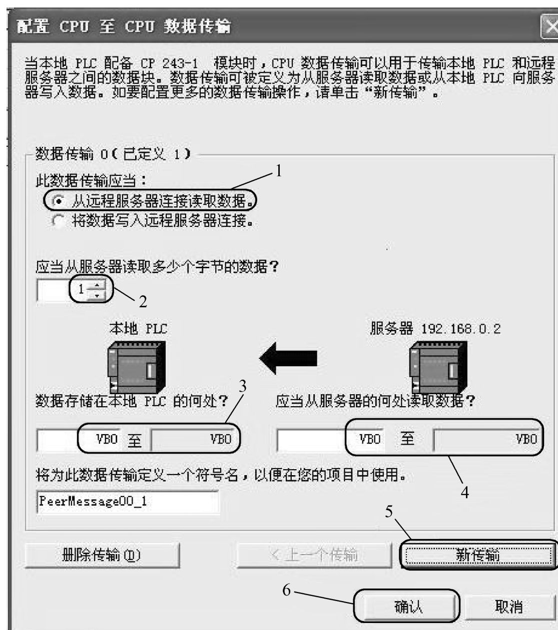


图 6-98　配置连接（3）

7）CRC 保护与保持现用间隔。如图 6-99 所示，先选定“是，为数据块中的此配置生成 CRC 保护”，再单击“下一步”按钮。

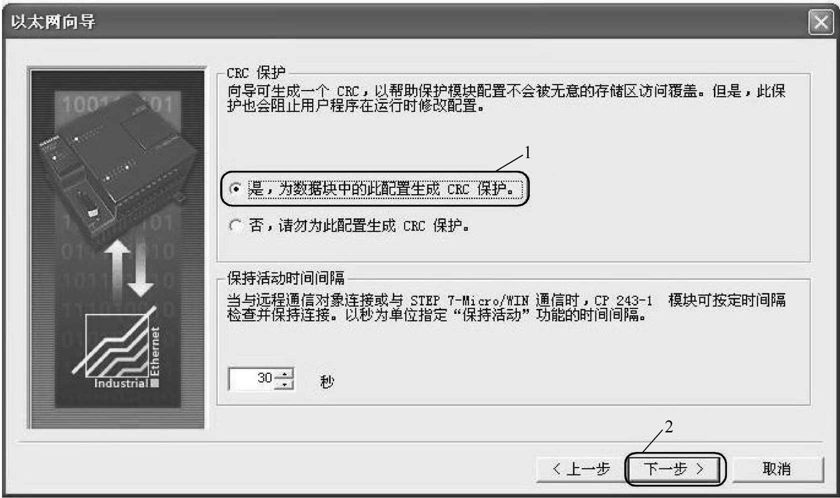


图 6-99　CRC 保护与保持现用间隔

8）分配配置内存。如图 6-100 所示，先单击“建议地址”按钮，再单击“下一步”按钮。



图 6-100　分配配置内存

9）生成项目部件。如图 6-101 所示，单击“完成”按钮，完成客户端的配置。

4. 编写程序

1）模块控制指令使用说明。ETHx_CTRL 指令是以太网向导自动生成的，它开始和执行以太网模块错误检查。应当在每次扫描开始调用该指令，且每个模块仅限使用一次。每次 CPU 更改为 RUN（运行）模式时，该指令命令 CP 243－1 以太网模块检查 V 内存区是否存在新配置。如果配置不同或 CRC 保护被禁止，则用新配置重设模块。当以太网模块准备从

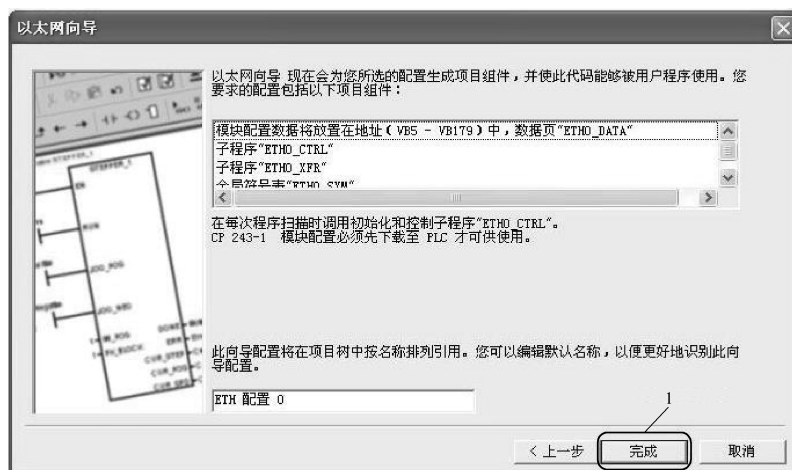


图 6-101　生成项目部件

其他指令接收命令时，CP_Ready变为现用。Ch_Ready 有一个指定给每个通道的位，显示该特定通道的连接状态。例如，当通道 0 建立连接后，位 0 打开。ETHx_CTRL 的指令格式见表 6-19。

表 6-19　ETHx_CTRL 的指令格式

LAD	输入/输出	说　明	数 据 类 型
ETH0_CTRL EN CP_Re~ Ch_Re~ Error	Error（错误）	错误代码	WORD
	CP_Ready	模块准备就绪	BOOL
	Ch_Ready	通道准备就绪	WORD

ETHx_XFR 指令也是以太网向导自动生成的，通过指定客户端连接和信息号码，命令在 S7－200 和远程连接之间进行数据传送。只有在选择至少配置一个客户端连接时，才会生成。数据传送所需的时间取决于使用的传输线路类型。如果希望提高传输速度，则应使用配备扫描时间低于 1 s 的程序。EN 位必须打开，才能启用模块命令，EN 位应当保持打开，直至设置表示执行完成的 Done（完成）位置位。当 START（开始）输入打开且模块目前不繁忙时，XFR 命令在每次扫描时均被发送至以太网模块。START（开始）输入可通过仅允许发送一条命令的边缘检测元素用脉冲方式打开。Chan_ID 是在向导中配置的一条客户端通道的号码。使用向导中指定的符号名。Data（数据）是为在向导中配置的指定信号定义的一个数据传送。使用向导中指定的符号名。Abort（异常中止）命令以太网模块停止在指定通道上的数据传送。该命令不会影响其他通道上的数据传送。如果指定通道的"保持现用"功能被禁用，当超出预期的超时限制时，则使用"异常中止"参数取消数据传送请求。当以太网模块完成数据传送时，Done（完成）打开。Error（错误）包含数据传送结果。ETHx_XFR（数据传输）的指令格式见表 6-20。

表 6-20　ETHx_XFR（数据传输）

LAD	输入/输出	说　　明	数 据 类 型
ETH0_XFR	EN	使能	BOOL
	START	如果模块不忙，向其发送开始命令	BOOL
	Chan_ID	客户机的通道 ID 号（0～7）	BYTE
	Data	要发送的信息 ID 号码（0～31）	BYTE
	Abort	异常中止	BOOL
	Done	完成时为 1	BOOL
	Error	错误代码	BYTE

2）编写程序。先编写服务器端的程序如图 6-102 所示，将程序下载到服务器端中。

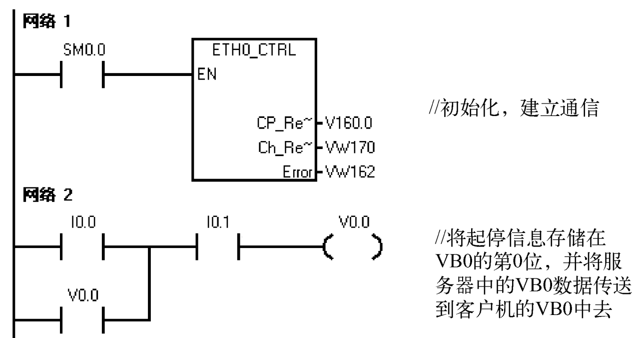


图 6-102　服务器端程序

再编写客户端的程序如图 6-103 所示，将程序下载到客户端中。

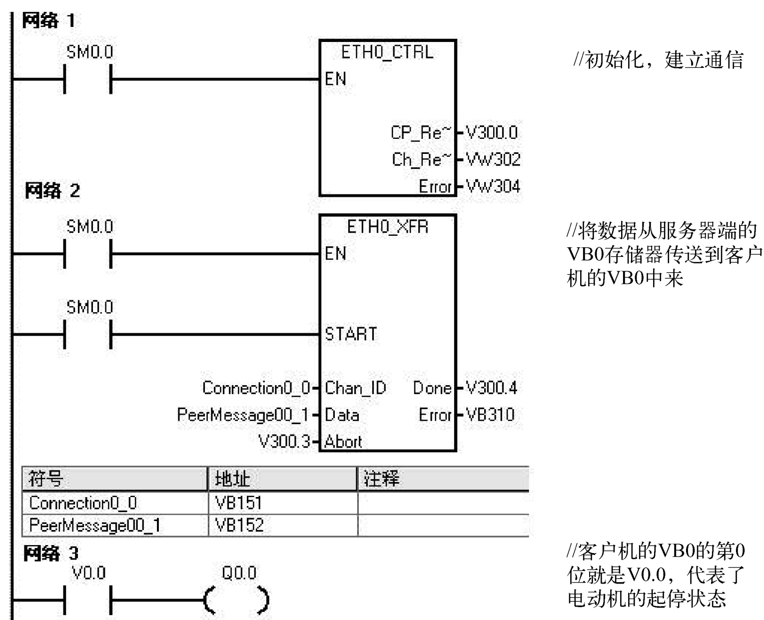


图 6-103　客户端程序

客户端通道号码（Chan_ID）的符号是 PeerMessage00_1，数据（Data）的符号是 Connectin0_0，这两个符号是以太网向导自动生成的，当打开如图 6-104 的符号表可知：PeerMessage00_1 的地址是 VB152，Connectin0_0 的地址是 VB151，编写程序时，只要将 VB151 写入客户端通道号码（Chan_ID）中，将自动弹出 Connectin0_0，将 VB152 写入数据（Data）中，将自动弹出 PeerMessage00_1。



图 6-104　符号表

3）程序运行结果。当服务器端和客户端运行时，若服务器端上的 I0.0 闭合，客户端上 Q0.0 控制的电动机起动；若服务器端上的 I0.1 闭合，客户端上 Q0.0 控制的电动机停止转动。

6.7　Modbus 通信

6.7.1　Modbus 通信概述

1. Modbus 协议简介

Modbus 协议是应用于电子控制器上的一种通用语言。通过此协议，控制器相互之间、控制器经由网络（例如以太网）和其他设备之间可以通信。它已经成为一种通用工业标准。有了它，不同厂商生产的控制设备可以连成工业网络，进行集中监控。

此协议定义了一个控制器能认识使用的消息结构，而不管它们是经过何种网络进行通信的。它描述了控制器请求访问其他设备的过程，如回应来自其他设备的请求，以及怎样侦测错误并记录。它制定了消息域格局和内容的公共格式。

当在一个 Modbus 网络上通信时，此协议决定了每个控制器须要知道它们的设备地址，识别按地址发来的消息，决定要产生何种行动。如果需要回应，控制器将生成反馈信息并用 Modbus 协议发出。在其他网络上，包含了 Modbus 协议的消息转换，在此网络上使用的帧或包结构。这种转换也扩展了根据具体的网络解决地址、路由路径及错误校验的方法。

2. Modbus 通信协议库

STEP 7-Micro/WIN 指令库包括专门为 Modbus 通信设计的预先定义的子程序和中断服

务程序，使得与 Modbus 设备的通信变得更简单。通过 Modbus 协议指令，可以将 S7 – 200 组态为 Modbus 主站或从站设备。

可以在 STEP 7 – Micro/WIN 指令树的库文件夹中找到这些指令。当在程序中输入一个 Modbus 指令时，自动将一个或多个相关的子程序添加到项目中。

西门子指令库以一个独立的光盘销售，在购买和安装了 1.1 版本的西门子指令库后，任何后续的 STEP 7 – Micro/WIN V3.2x 和 V4.0 升级都会在不需要附加费用的情况下自动升级指令库（当增加或修改库时）。

【关键点】 STEP 7 – Micro/WIN V4.0 SP4（含）以前的版本，指令库只有从站指令，之后的版本才有主站指令库，如果需要 SP4（含）以前 S7 – 200 做主站，读者必须在自由口模式下，按照 Modbus 协议编写程序，这会很麻烦。CPU 的固化程序版本不低于 V2.0 才能支持 Modbus 指令库。

3. Modbus 的地址

Modbus 地址通常是包含数据类型和偏移量的 5 个字符值。第一个字符确定数据类型，后面四个字符选择数据类型内的正确数值。

（1）主站寻址

Modbus 主站指令可将地址映射到正确功能，然后发送至从站设备。Modbus 主站指令支持下列 Modbus 地址：

- 00001 ~ 09999 是离散输出（线圈）。
- 10001 ~ 19999 是离散输入（触点）。
- 30001 ~ 39999 是输入寄存器（通常是模拟量输入）。
- 40001 ~ 49999 是保持寄存器。

所有 Modbus 地址都是基于 1，即从地址 1 开始第一个数据值。有效地址范围取决于从站设备。不同的从站设备将支持不同的数据类型和地址范围。

（2）从站寻址

Modbus 主站设备将地址映射到正确功能。Modbus 从站指令支持以下地址：

- 00001 ~ 00128 是实际输出，对应于 Q0.0 ~ Q15.7。
- 10001 ~ 10128 是实际输入，对应于 I0.0 ~ I15.7。
- 30001 ~ 30032 是模拟输入寄存器，对应于 AIW0 ~ AIW62。
- 40001 ~ 04XXXX 是保持寄存器，对应于 V 区。

所有 Modbus 地址都是从 1 开始编号的。Modbus 地址与 S7 – 200 地址的对应关系见表 6-21。

表 6-21　Modbus 地址与 S7 – 200 地址的对应关系

序　　号	Modbus 地址	S7 – 200 地址
1	00001	Q0.0
	00002	Q0.1
	…	…
	00127	Q15.6
	00128	Q15.7

（续）

序　　号	Modbus 地址	S7－200 地址
2	10001	I0.0
	10002	I0.1
	…	…
	10127	I15.6
	10128	I15.7
3	30001	AIW0
	30002	AIW1
	…	…
	30032	AIW62
4	40001	HoldStart
	40002	HoldStart＋2
	…	
	4xxxx	HoldStart＋2×(xxxx－1)

Modbus 从站协议允许对 Modbus 主站可访问的输入、输出、模拟输入和保持寄存器（V 区）的数量进行限定。例如，若 HoldStart 是 VB0，那么 Modbus 地址 40001 对应 S7－200 地址的 VB0。

6.7.2　S7－200 PLC Modbus 通信的应用

下面以两台 CPU 226CN 之间的 Modbus 现场总线通信为例介绍 S7－200 系列 PLC 之间的 Modbus 现场总线通信。

【例 6-9】 模块化生产线的主站为 CPU 226CN，从站为 CPU 226CN，主站发出开始信号（开始信号为高电平），从站接收信息，并控制从站的电动机的起停。

【解】

1. 主要软硬件配置

① 1 套 STEP 7－Micro/WIN V4.0 SP9。

② 1 根 PC/PPI 电缆（或者 CP 5611 卡）。

③ 2 台 CPU 226CN。

④ 1 根 PROFIBUS 网络电缆（含两个网络总线连接器）。

Modbus 现场总线硬件配置如图 6-105 所示。

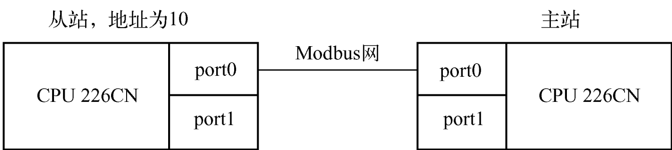


图 6-105　Modbus 现场总线硬件配置图

2. 相关指令介绍

（1） 主设备指令

初始化主设备指令 MBUS_CTRL 用于 S7－200 端口 0（或用于端口 1 的 MBUS_CTRL_P1 指令）可初始化、监视或禁用 Modbus 通信。在使用 MBUS_MSG 指令之前，必须正确执行 MBUS_CTRL 指令，指令执行完成后，立即设定“完成”位，才能继续执行下一条指令。其各输入/输出参数见表 6-22。

表 6-22　MBUS_CTRL 各输入/输出参数表

子　程　序	输入/输出	说　　明	数 据 类 型
MBUS_CTRL EN Mode Baud　Done Parity　Error Timeout	EN	使能	BOOL
	Mode	为 1 将 CPU 端口分配给 Modbus 协议并启用该协议。为 0 将 CPU 端口分配给 PPI 协议，并禁用 Modbus 协议	BOOL
	Baud	将波特率设为 1200、2400、4800、9600、19200、38400、57600 或 115200 波特	D WORD
	Parity	0－无奇偶校验　1－奇校验　2－偶校验	BYTE
	Timeout	等待来自从站应答的毫秒时间数	WORD
	ERROR	出错时返回错误代码	BYTE

MBUS_MSG 指令（或用于端口 1 的 MBUS_MSG_P1）用于启动对 Modbus 从站的请求，并处理应答。当 EN 输入和“首次”输入打开时，MBUS_MSG 指令启动对 Modbus 从站的请求。发送请求、等待应答、并处理应答。EN 输入必须打开，以启用请求的发送，并保持打开，直到“完成”位被置位。此指令在一个程序中可以执行多次。其各输入/输出参数见表 6-23。

表 6-23　MBUS_MSG 各输入/输出参数表

子　程　序	输入/输出	说　　明	数 据 类 型
MBUS_MSG EN First Slave　Done RW　Error Addr Count DataPtr	EN	使能	BOOL
	First	“首次”参数应该在有新请求要发送时才打开，进行一次扫描。“首次”输入应当通过一个边沿检测元素（例如上升沿）打开，这将保证请求被传送一次	BOOL
	Slave	“从站”参数是 Modbus 从站的地址。允许的范围是 0～247	BYTE
	RW	0－读，1－写	BYTE
	Addr	“地址”参数是 Modbus 的起始地址	DWORD
	Count	“计数”参数，读取或写入的数据元素的数目	INT
	DataPtr	S7－200 CPU 的 V 存储器中与读取或写入请求相关数据的间接地址指针	DWORD
	ERROR	出错时返回错误代码	BYTE

【**关键点**】指令 MBUS_CTRL 的 EN 要接通，在程序中只能调用一次，MBUS_MSG 指令可以在程序中多次调用，要特别注意区分 Addr 、DataPtr 和 Slave 三个参数。

（2）从设备指令

MBUS_INIT 指令用于启用、初始化或禁止 Modbus 通信。在使用 MBUS_SLAVE 指令之前，必须正确执行 MBUS_INIT 指令。指令完成后立即设定“完成”位，才能继续执行下一条指令。其各输入/输出参数见表 6-24。

表 6-24　MBUS_INIT 各输入/输出参数表

子　程　序	输入/输出	说　　明	数 据 类 型
MBUS_INIT EN Mode　Done Addr　Error Baud Parity Delay MaxIQ MaxAI MaxHold HoldSt~	EN	使能	BOOL
	Mode	为 1 将 CPU 端口分配给 Modbus 协议并启用该协议。为 0 将 CPU 端口分配给 PPI 协议，并禁用 Modbus 协议	BYTE
	Baud	将波特率设为 1200、2400、4800、9600、19200、38400、57600 或 115200	DWORD
	Parity	0－无奇偶校验，1－奇校验，2－偶校验	BYTE
	Addr	“地址”参数是 Modbus 的起始地址	BYTE
	Delay	“延时”参数，通过将指定的毫秒数增加至标准 Modbus 信息超时的方法，延长标准 Modbus 信息结束超时条件	WORD
	MaxIQ	参数将 Modbus 地址 0xxxx 和 1xxxx 使用的 I 和 Q 点数设为 0～128 之间的数值	WORD
	MaxAI	参数将 Modbus 地址 3xxxx 使用的字输入（AI）寄存器数目设为 0～32 之间的数值。	WORD
	MaxHold	参数设定 Modbus 地址 4xxxx 使用的 V 存储器中的字保持寄存器数目	WORD
	HoldStart	参数是 V 存储器中保持寄存器的起始地址	DWORD
	ERROR	出错时返回错误代码	BYTE

MBUS_SLAVE 指令用于为 Modbus 主设备发出的请求服务，并且必须在每次扫描时执行，以便允许该指令检查和回答 Modbus 请求。在每次扫描且 EN 输入开启时，执行该指令。其各输入/输出参数见表 6-25。

表 6-25　MBUS_SLAVE 各输入/输出参数表

子　程　序	输入/输出	说　　明	数 据 类 型
MBUS_SLAVE EN Done Error	EN	使能	BOOL
	Done	当 MBUS_SLAVE 指令对 Modbus 请求作出应答时，“完成”输出打开。如果没有需要服务的请求时，“完成”输出关闭	BOOL
	ERROR	出错时返回错误代码	BYTE

【关键点】MBUS_INIT 指令只在首次扫描时执行一次，MBUS_SLAVE 指令无输入参数。

3. 编写程序

主站和从站的程序如图 6-106 和图 6-107 所示。

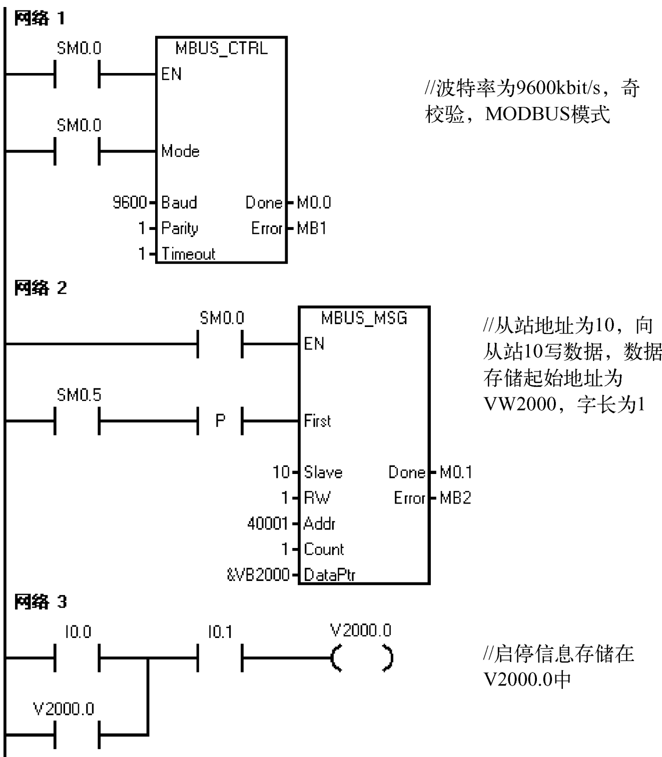


图 6-106　主站程序

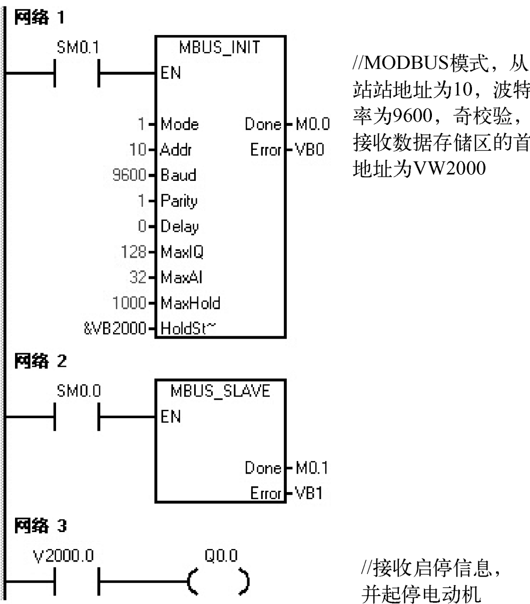


图 6-107　从站程序

【关键点】在调用了 Modbus 指令库的指令后，还要对库存储区进行分配，这是非常重要的，否则即使编写程序没有语法错误，程序编译后也会显示至少几十个错误。分配库存储区的方法如下：先选中“程序块”，再单击鼠标右键，弹出快捷菜单，并单击“库存储区”，如图 6-108 所示。再在“库存储区”中填写 Modbus 指令所需要用到的存储区的起始地址，如图 6-109 所示。示例中 Modbus 指令所需要用到的存储区为 VB0 ~ VB283，这个区间的 V 存储区在后续编程是不能使用的。

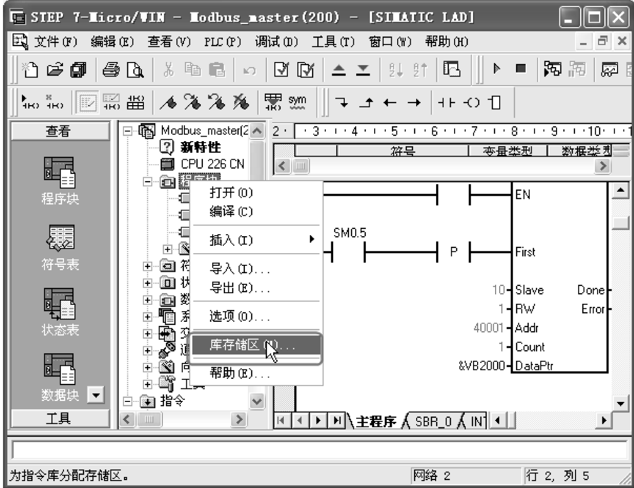


图 6-108　选定库存储区

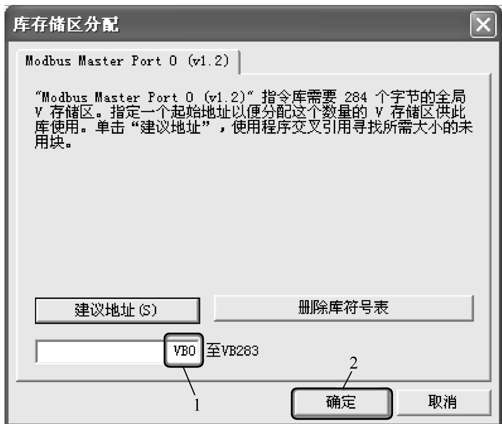


图 6-109　设定库存储区的范围

小　　结

重点难点总结

通信是 PLC 中的难点，一般初学者不容易掌握，往往通信不成功时，还不知道错误出在何处。其实西门子的每种通信都有固定的模式，只要掌握正确步骤，就可以完成通信。掌握西门子 PLC 的通信后，在学习其他品牌的 PLC 通信，就会很容易掌握了，因为通信的机

理都是一样的。

（1）PROFIBUS 现场总线应用很广泛，应重点掌握。

（2）工业以太网在稍大的系统中都有应用，特别是车间级及以上的通信系统中应用较多，应该重点掌握。

（3）PROFIBUS 现场总线和工业以太网通信的难点有两点：一是通信的组态，步骤较多，初学时容易出错；二是通信专用函数的参数较多，不易理解。

因此要学好本章内容必须要下功夫。

习　　题

1. OSI 模型分为哪几个层？各层的作用是什么？

2. PROFIBUS 现场总线的种类有哪些？这些种类的 PROFIBUS 现场总线使用了 OSI 模型的哪几层？

3. 西门子 PLC 的常见通信方式有哪几种？

4. 有三台 CPU 226CN，一台为主站，其余两台为从站，在主站上发出一个起停信号，对从站上控制的电动机进行起停，从站将电动机的起停状态反馈到主站，请用网络读/写指令编写程序。

5. 有三台 CPU 226CN，一台为主站，其余两台为从站，在主站上发出一个起停信号，对从站上控制的电动机进行起停，从站将电动机的起停状态反馈到主站，请用指令向导生成子程序，并编写程序。

6. 某设备上有一台 CPU 314C－2DP 为主站，有两台 CPU 226CN 为从站，在主站上发出一个起停信号，对从站上控制的电动机进行起停，从站将电动机的起停状态反馈到主站，请组态硬件并编写程序。

7. 何谓串行通信和并行通信？

8. 何谓双工、单工和半双工？请举例说明。

第 7 章　上位机对 S7 – 200 系列 PLC 的监控

本章介绍了组态软件的功能，以及怎样在上位机中用组态软件监控 S7 – 200 PLC。

7.1　简单组态软件工程的建立

7.1.1　认识组态软件

1. 初识组态软件

在使用工控软件时，人们经常提到“组态”一词，组态的英文是“Configuration”，简而言之，组态就是利用应用软件中提供的工具、方法完成工程中某一具体任务的过程。组态软件是数据采集监控（Supervisory Control and Data Acquisition，SCANDA）系统的软件平台工具，是工业应用软件的一个组成部分。它具有丰富的项目设置，使用方式灵活，功能强大。组态软件由早先单一的人机界面向数据处理方向发展，管理的数据量越来越大。随着组态软件自身以及控制系统的发展，监控组态软件部分与硬件分离，为自动化软件的发展提供了充分发挥作用的舞台。OPC（OLE for Process Control）的出现，以及现场总线和工业以太网的快速发展，大大简化了不同厂家设备之间的互连，降低了开发 I/O 设备驱动软件的工作量。

实时数据库的作用进一步加强。实时数据库是 SCANDA 系统的核心技术。从软件技术上讲，SCANDA 系统的实时数据库，实际上就是一个可统一管理、支持变结构、支持实时计算的数据结构模型。

社会信息化的加速发展是组态软件市场增长的强大推动力。在最终用户眼里，组态软件在自动化系统中发挥的作用逐渐增大，甚至有时到了非用不可的地步。主要原因在于：组态软件的功能强大，用户普遍需求，而且逐渐认识其价值。

2. 组态软件的功能

组态软件采用类似资源浏览器的窗口结构，并对工业控制系统中的各种资源（设备、标签量和画面等）进行配置和编辑；处理数据报警和系统报警；提供多种数据驱动程序；各类报表的生成和打印输出；使用脚本语言提供二次开发功能；存储历史数据，并支持历史数据的查询等。

3. 组态软件的发展趋势

新技术在组态软件中的应用，使得组态软件呈现如下发展趋势：

① 多数组态软件提供多种数据采集驱动程序（driver），用户可以进行配置。驱动程序通常由组态软件开发商提供，并按照某种规范编写。

② 脚本语言是扩充组态系统功能的重要手段。脚本语言大体有两种形式，一是 C/BASIC语言，二是微软的 VBA 编程语言。

③ 具备二次开发的能力。在不改变原来系统的情况下，向系统增加新功能的能力。增

加新功能最常用的就是 ActiveX 组件的应用。

④ 组态软件的应用具有高度的开放性。

⑤ 与 MES（Manufacturing Execution System）和 ERP（Enterprise Resource Planing）系统紧密集成。

⑥ 现代企业的生产已经趋向国际化、分布式的生产方式。互联网是实现分布式生产的基础。组态软件将原来的局域网运行方式跨越到支持 Internet。

4. 常用的组态软件简介

1）InTouch。它是最早进入我国的组态软件。早期的版本采用 DDE（动态数据交换）方式与驱动程序通信，性能较差。新的版本采用了 32 位 Windows 平台，并提供 OPC 支持。

2）iFIX。它是 Intellution 公司起家时开发的软件，后被爱默生公司收购，现在又被 GE 公司收购。iFIX 的功能强大，使用比较复杂。iFIX 驱动程序和 OPC 组件需要单独购买。iF-IX 的价格比较昂贵。

3）Citech。澳大利亚 CiT 公司的 Citech 是较早进入中国市场的产品。Citech 的优点是操作方式简洁，但脚本语言比较麻烦，不易掌握。

4）WinCC。Siemens 公司的 WinCC 是后起之秀，1996 年才进入市场，当年就被美国的 Control Engineering 杂志评为当年的最佳 HMI 软件。它是一套完备的组态开发环境，内嵌 OPC。WinCC V6.0 采用 Microsoft SQL Server 2000 数据库进行生产数据存档，同时它具有 Web 服务器功能。

5）组态王。组态王是北京亚控公司的产品，是国产组态软件的代表，在国内有一定的市场。组态王提供了资源管理器式的操作界面，并且提供以汉字为关键字的脚本语言支持，这点是国外组态软件很难做到的。另外，组态王提供了丰富的国内外硬件设备驱动程序，这点国外知名组态软件也很难做到。

6）三维力控。三维力控是国内较早开发成功的组态软件，其最大的特点就是基于真正意义的分布式实时数据库的三层结构，而且实时数据库是可组态的。三维力控组态软件也提供了丰富的国内外硬件设备驱动程序。

另外，国内外比较有名的组态软件还有 GE 的 Cimplicity、华富计算机公司的开物和北京昆仑通态的 MCGS 等。总之，在国内，一般比较大型的控制系统多用国外的组态软件，而在中低端市场，国产组态软件则有一定的优势。

7.1.2 安装 WinCC 的硬件要求

完整 WinCC V7.0 SP1 软件比 WinCC V6.0 的容量要大得多，所以其对软硬件的要求比较高，见表 7-1。

表 7-1 硬件的要求

硬　件	操作系统	最小值	推荐值
CPU	Windows XP	客户机：Intel Pentium III；800 MHz 单用户系统：Intel Pentium III；1 GHz	客户机：Intel Pentium 4；2 GHz 单用户系统：Intel Pentium 4；2.5 GHz

（续）

硬　件	操作系统	最小值	推荐值
CPU	Windows Vista	客户机：Intel Pentium 4；2.5 GHz 单用户系统：Intel Pentium 4；2.5 GHz	客户机：Intel Pentium 4；3 GHz/双核 单用户系统：Intel Pentium 4；3.5 GHz/双核
	Windows Server 2003	单用户系统：Intel Pentium III；1 GHz 服务器：Intel Pentium III；1 GHz 中央归档服务器：Intel Pentium 4；2.5 GHz	单用户系统：Intel Pentium 4；3 GHz 服务器：Intel Pentium 4；3 GHz 中央归档服务器：Intel Pentium 4；3 GHz/双核
工作内存	Windows XP	客户机：512 MB 单用户系统：1 GB	客户机：≥1 GB 单用户系统：2 GB
	Windows Vista	客户机：1 GB 单用户系统：2 GB	客户机：2 GB 单用户系统：2 GB
	Windows Server 2003	单用户系统：1 GB 服务器：1 GB 中央归档服务器：2 GB	单用户系统：2 GB 服务器：2 GB 中央归档服务器：>2 GB
硬盘上的可用内存 - 用于安装 WinCC - 用于使用 WinCC		客户机：1.5 GB/服务器：>1.5 GB 客户机：1.5 GB/服务器：2 GB/中央归档服务器：40 GB	客户机：>1.5 GB/服务器：2 GB 客户机：>1.5 GB/服务器：10 GB/中央归档服务器：不同硬盘上有 2 个各为 80 GB 的可用空间
虚拟内存		1.5 倍工作内存	1.5 倍工作内存
Windows 打印机假脱机程序内存		100 MB	>100 MB
图形卡		16 MB	32 MB
颜色深度/颜色质量		256	最高（32 位）
分辨率		800×600	1024×768

7.1.3　安装 WinCC 的软件要求

1. 操作系统

（1）支持语言

- WinCC 的欧洲版支持德语、英语、法语、意大利语和西班牙语。
- WinCC 的亚洲版支持简体中文、繁体中文、日语和朝鲜语。

（2）单用户系统和客户机

- Windows XP Professional Service Pack 2 或 Service Pack 3。
- Windows Vista Business Service Pack 1、Vista Enterprise Service Pack 1、Vista Ultimate Service Pack1。

- Windows Server 2003。

（3） WinCC 服务器

- 可在 Windows Server 2003 标准版/企业版或 Windows Server 2003 R2 上操作 WinCC 服务器。
- 如果正在运行的客户机不超过三个，也可以在 Windows XP 上操作 WinCC Runtime Server。针对此组态的 WinCC Service Mode 尚未发布。

Windows Vista 不允许作为 WinCC 服务器运行。

2. Microsoft 消息队列服务

WinCC 需要 Microsoft 消息队列服务。

3. Microsoft SQL Server 2005

WinCC 要求有 Microsoft SQL Server 2005 SP2。将在安装 WinCC 期间自动安装 SQL 服务器。随 Microsoft SQL Server 2005 还安装必需的连通性组件。

4. Internet Explorer 的要求

至少 Microsoft Internet Explorer V6.0 Service Pack 1 以上版本。

7.2 用 WinCC 监控 S7 - 200 PLC

WinCC 软件中只有少数驱动程序，也就是说 WinCC 软件自带的驱动程序只能和 S7 - 300/400 等少数几种 PLC 直接进行通信。即使是西门子的 S7 - 200 和 S7 - 1200，WinCC 软件中也没有直接与之通信的驱动程序。那么是不是 WinCC 软件不能与 S7 - 200 通信呢？其实要建立 WinCC 软件与 S7 - 200 通信，需要借助于 OPC。常见的可以用于 S7 - 200 的 OPC 软件有西门子的 SIMATIC NET 和 PC Access 软件，此外还有 Kepserver，当然高水平的技术人员还可以自己编写通信程序。

7.2.1 OPC 基本知识

1. OPC 概念

在 OPC 之前，需要花费很多时间使用软件应用程序控制不同供应商的硬件。存在多种不同的系统和协议；用户必须为每一家供应商和每一种协议订购特殊的软件，才能存/取具体的接口和驱动程序。因此，用户程序取决于供应商、协议或系统。而 OPC 具有统一和非专有的软件接口，在自动化工程中具有强大的数据交换功能。

OPC（OLE for Process Control）是嵌入式过程控制标准，规范以 OLE/DCOM 为技术基础，是用于服务器/客户机连接的统一而开放的接口标准和技术规范。OLE 是微软为Windows系统、应用程序间的数据交换而开发的技术，是“Object Linking and Embedding”的缩写。

OPC 从数据来源地提供数据并以标准方式将数据传输至任何客户机应用程序的机制。供应商现在能够开发一种可重新使用、高度优化的服务器，与数据来源通信，并保持从数据来源/设备有效地存/取数据的机制。为服务器提供 OPC 接口允许任何客户机存/取设备。

OPC（用于进程控制的 OLE）是一种开放式系统接口标准，可允许在自动化/PLC 应用、现场设备和基于 PC 的应用程序（例如 HMI 或办公室应用程序）之间进行简单的标准化数据交换。定义工业环境中各种不同应用程序的信息交换，它工作于应用程序的下方。您可以

在 PC 上监控、调用和处理可编程序控制器的数据和事件。

2. 服务器与客户机的概念

OPC 服务器和客户机的概念与超级市场相似，存放各种供选择商品的通道代表服务器。供选择的商品构成服务器读取和写入的所有进程数据位置。客户机就如同沿着通道移动并选择需要物品的购物车。OPC 数据项是 OPC 服务器与数据来源的连接。所有与 OPC 数据项的读/写、存/取均通过包含 OPC 项目的 OPC 群组目标进行。同一个 OPC 项目可包含在几个群组中。当某个变量被查询时，对应的数值会从最新进程数据中获取并被返回，这些数值可以是传感器、控制参数、状态信息或网络连接状态的数值。OPC 的结构由 3 类对象组成：服务器、组和数据项。

OPC 服务器：提供数据的 OPC 元件被称为 OPC 服务器。OPC 服务器向下对设备数据进行采集，向上与 OPC 客户应用程序通信完成数据交换。

OPC 客户机：使用 OPC 服务器作为数据源的 OPC 元件称为 OPC 客户机。

3. OPC 数据访问

OPC 服务器支持两种类型的数据读/取：同步读/写（Synchronous read/write）和异步读/写（Asynchronous read/write）。

同步读/写：OPC 的客户机向服务器发出一个读/写请求，然后不再继续执行，一直等待直到收到服务器发给客户机的返回值，OPC 客户机才会继续执行下去。

异步读/写：OPC 的客户机向服务器发出一个读/写请求，在等待返回值的过程中，可以继续执行下面的程序，直到服务器数据准备好后，向客户机发出一个返回值，在回调函数中客户机处理返回数值，然后结束此次读/写过程。

同步读/写数据存/取速度快，编程简单，无需回调，但需要等待返回结果。异步读/写不需等待返回值，可以同时处理多个请求。

7.2.2 PC Access 软件简介

PC Access 软件是西门子推出的专用于 S7－200 PLC 的 OPC Server（服务器）软件，它向 OPC 客户机提供数据信息，可以与任何标准的 OPC Client（客户机）通信。PC Access 软件自带 OPC 客户测试端，用户可以方便的检测其项目的通信及配置的正确性。

PC Access 可以用于连接西门子，或者第三方的支持 OPC 技术的上位机软件。

1. PC Access 的兼容性

① 支持 OPC Data Access（DA）3.0 版（Version 3.0）。

② 可以运行在 Windows 2000 或 Windows XP。

③ 可以从 Micro/WIN 项目（V3.x～V4.x）中导入符号表。

④ 支持新的 S7－200 智能电缆（RS－232 或 USB）。

⑤ 支持多种语言：英语、中文、德语、法语、意大利语、西班牙语。

PC Access 的升级包可以在 S7－200 产品主页上免费下载、安装。下载地址链接：http://support.automation.siemens.com/WW/view/en/18785011/133100。

2. PC Access 安装的要求

1）PC Access 可以在 Microsoft 的如下操作系统中安装、使用：

- Windows 2000 SP3 以上。

- Windows XP Home。
- Windows XP Professional。

2）PC 的硬件要求如下：

- 任何可以安装运行上述操作系统的计算机。
- 最少 150 MB 硬盘空间。
- Microsoft Windows 支持的鼠标。
- 推荐使用 1024×768 像素的屏幕分辨率，小字体。

3. PC Access 支持的硬件连接

PC Access 可以通过如下硬件连接与 S7－200 通信：

- 通过 PC/PPI 电缆（USB/PPI 电缆）连接 PC 上的 USB 口和 S7－200。
- 通过 PC/PPI 电缆（RS－232/PPI 电缆）连接 PC 上的串行 COM 口和 S7－200。
- 通过西门子通信处理器（CP）卡和 MPI 电缆连接 S7－200。
- 通过 PC 上安装的调制解调器（Modem）连接 S7－200 上的 EM 241 模块。
- 通过以太网连接 S7－200 上的 CP 243－1 或 CP 243－1 IT 模块。

上述 S7－200 的通信口可以是 CPU 通信口，也可以是 EM 277 的通信口。

【关键点】 PC Access 不支持 CP 5613 和 CP 5614 通信卡。

4. PC Access 的协议连接

1）PC Access 所支持的协议如下：

- PPI（通过 RS－232PPI 和 USB/PPI 电缆）。
- MPI（通过相关的 CP 卡）。
- PROFIBUS－DP（通过 CP 卡）。
- S7 协议（以太网）。
- Modems（内部的或外部的，使用 TAPI 驱动器）。

2）所有协议允许同时有 8 个 PLC 连接。

3）一个 PLC 通信口允许有 4 个 PC 的连接，其中一个连接预留给 STEP 7－Micro/WIN。

4）PC Access 与 STEP 7－Micro/WIN 可以同时访问 CPU。

5）支持 S7－200 所有内存数据类型。

5. PC Access 的特性

- 内置的 OPC 测试 Client 端，直接将 Item 中的数据标签拖入 Test Client 窗口中，并单击工具栏中的 Test Client Status 按钮即可监测数据。
- 可以添加 Excel Client 端，用于简单的电子表格对 S7－200 数据的监控。
- 提供任何 OPC Client 端的标准接口。
- 针对于每一标签刷新的时间戳。

6. PC Access 技术要点

- 不能直接访问 PLC 存储卡中的信息（数据归档、配方）。
- 不包含用于创建 VB 客户端的控件。
- 可以在 PC 上用 STEP 7－Micro/WIN 4.0 和 PC Access 同时访问 PLC（必须使用同一种通信方式）。
- 在同一 PC 上不能同时使用 PC/PPI 电缆、Modem 或 Ethernet 访问同一个或不同的

PLC，只支持 PG/PC－Interface 中所设置的单一的通信方式。

- PC Access 中没有打印工具。
- 使用同一通信通道，最多可以同时监控 8 个 PLC。
- Item 的个数没有限制。
- 可应用于当前 Siemens 提供的所有 CP 卡。
- PC Access 专为 S7－200 而设计，不能应用于 S7－300 或 S7－400。

7.2.3 WinCC 与 S7－200 PLC 的通信

WinCC 中没有提供 S7－200 系列 PLC 的驱动程序，要用 WinCC 对 S7－200 PLC 进行监控，必须使用 OPC 通信，以下用一个简单的例子，讲解这个过程。

【例 7-1】 WinCC 对 S7－200 PLC 进行监控，在 WinCC 画面上启动和停止 S7－200 PLC 的一盏灯，并将灯明暗状态显示在 WinCC 画面上。

【解】

1. 所需要的软硬件

- 1 套 S7－200 PC Access V3.0。
- 1 套 STEP 7－Micro/Win V4.0 SP9。
- 1 套 WinCC V7.0 SP1。
- 1 台 CPU 226CN。
- 1 根 PC/PPI 电缆。
- 1 台个人计算机（具备安装和运行 WinCC V7.0 SP1 的条件）。

2. 操作步骤

1）在 S7－200 PC Access 中创建 OPC。具体步骤如下：

① 新建项目。打开 S7－200 PC Access 软件（此软件可以在西门子的官网上免费下载。如果不能直接下载，可以和西门子公司的技术支持人员联系，西门子公司的技术支持人员会发给读者一个下载该软件的链接），新建项目如图 7-1 所示。

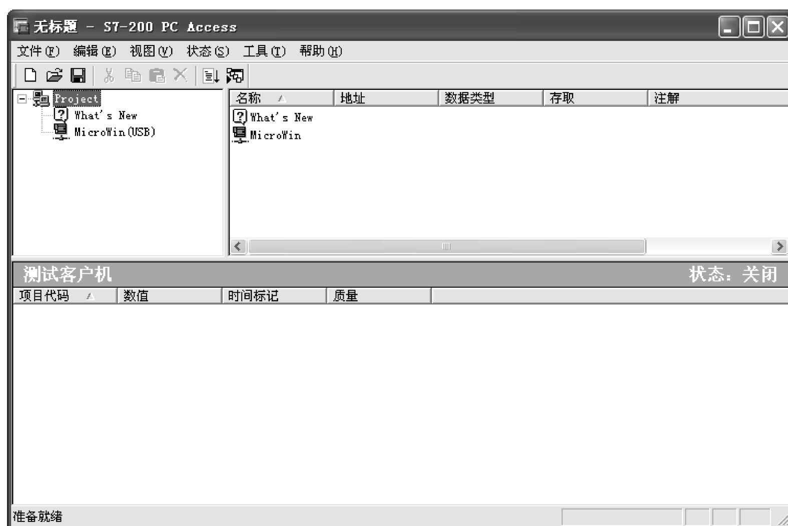


图 7-1　新建项目

② 新建 PLC。在左侧的浏览器窗口中，如图 7-2 所示，选中“MicroWin（USB）”，单击鼠标右键，弹出快捷菜单，单击“新 PLC”选项，弹出“PLC 属性”界面，如图 7-3 所示。将 PLC 命名为“S7 - 200”，单击“确认”按钮。



图 7-2 新建 PLC



图 7-3 PLC 属性

③ 新建变量。在左侧的浏览器窗口中，选择以上步骤中创建的 PLC“S7 - 200”，单击鼠标右键弹出快捷菜单，单击“新”→“项目”选项，如图 7-4 所示。



图 7-4 新建变量（1）

如图 7-5 所示，在名称中输入“START”，在地址中输入“M0.0”，最后单击“确认”按钮。这样做使变量“START”的地址是“M0.0”。用同样的方法操作，使变量“STOP”的地址是“M0.1”，使变量“MOTOR”的地址是“Q0.0”。操作完成后所有的变量和地址都显示在如图 7-6 所示的界面上。



图 7-5　新建变量（2）



图 7-6　新建变量（3）

④ 保存 OPC。单击工具栏中的“保存”按钮，弹出如图 7-7 所示的界面，命名为“S7 - 200. pca”，单击“保存”按钮。

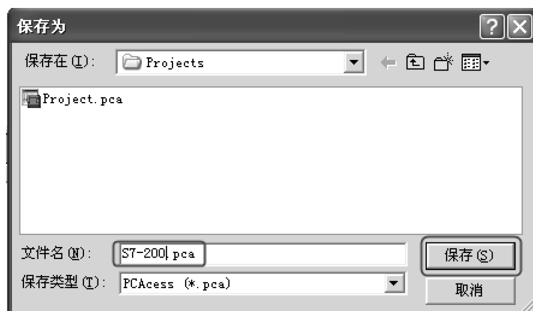


图 7-7　保存变量

2）在 WinCC 中创建工程，完成通信。具体步骤如下：

① 新建工程。单击工具栏上的“新建”图标 ，弹出如图 7-8 所示的界面，将项目名称定为“S7200”，单击“创建”按钮。



图 7-8　新建工程

② 添加驱动程序。如图 7-9 所示，选中左侧的浏览器窗口的变量管理器，单击鼠标右键，弹出快捷菜单，单击“添加新的驱动程序”，弹出如图 7-10 所示界面，选中“OPC. chn”，单击“打开”按钮。

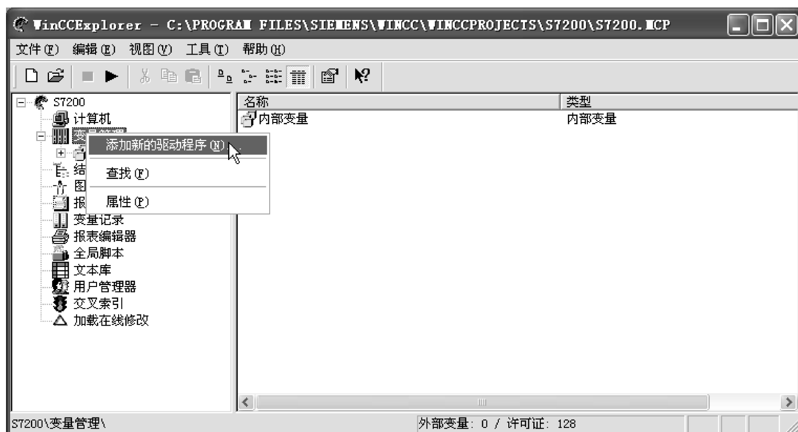


图 7-9　添加驱动程序（1）

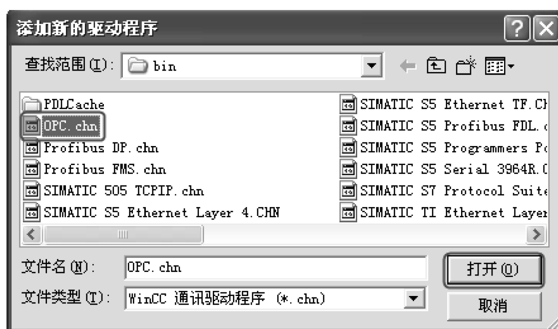


图 7-10　添加驱动程序（2）

③ 打开系统参数。如图 7-11 所示，选中左侧的浏览器窗口的“OPC Group…”，单击鼠标右键，弹出快捷菜单，单击“系统参数”，弹出如图 7-12 所示界面，选中“S7200. OPCServer”，单击“浏览服务器”按钮。

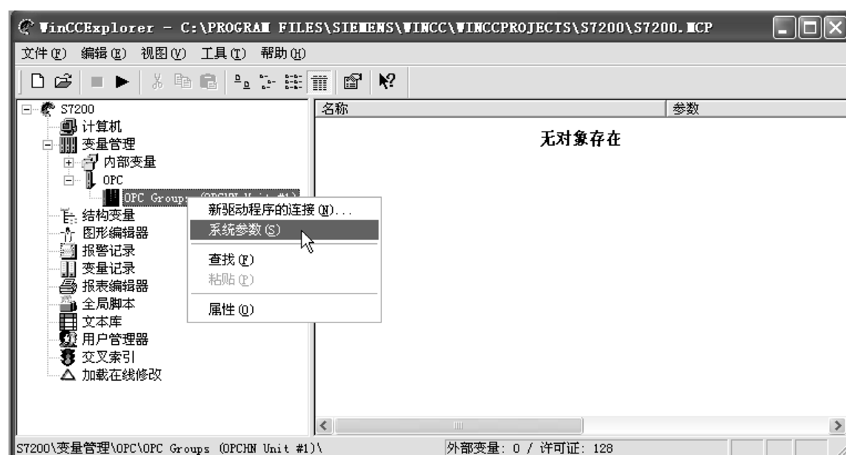


图 7-11　打开系统参数

如图 7-13 所示，单击“下一步”按钮，弹出如图 7-14 所示的界面，单击“添加项目”按钮。

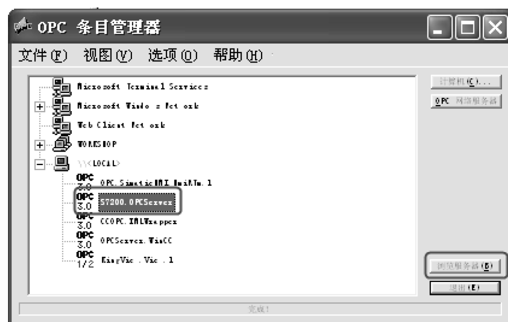


图 7-12　OPC 项目管理器



图 7-13　过滤标准

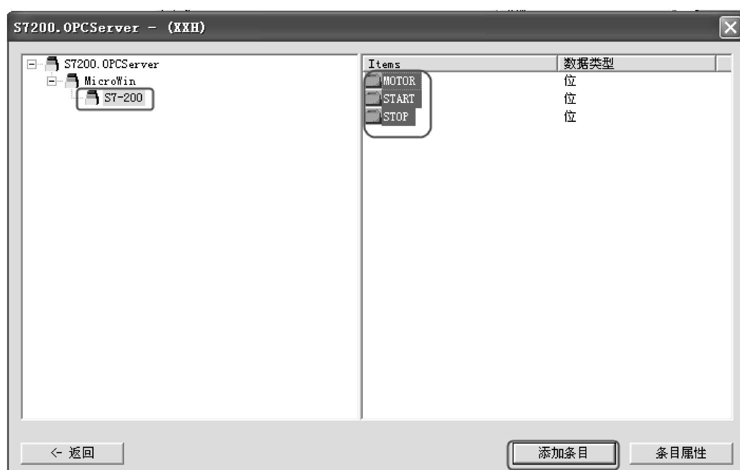


图 7-14　添加项目

④ 添加连接。单击“是”按钮，如图 7-15 所示，弹出如图 7-16 所示的界面，输入连接名称为“S7200_OPCServer”，单击“确定”按钮。

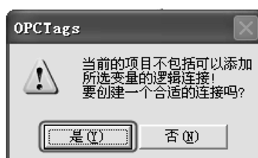


图 7-15　添加连接（1）



图 7-16　添加连接（2）

⑤ 添加变量。如图 7-17 所示，单击“完成”按钮即可。变量添加完成后，如图 7-18 所示，在 PC Access 中创建的变量，在 WinCC 中都可以搜索到。



图 7-17　添加变量（1）

⑥ 动画链接。在图形编辑器中，拖入一个圆，选中此圆，双击之，弹出“对象属性”，选中“属性”选项卡，接着选中“效果”→“全局颜色方案”，把选项“是”改为“否”，

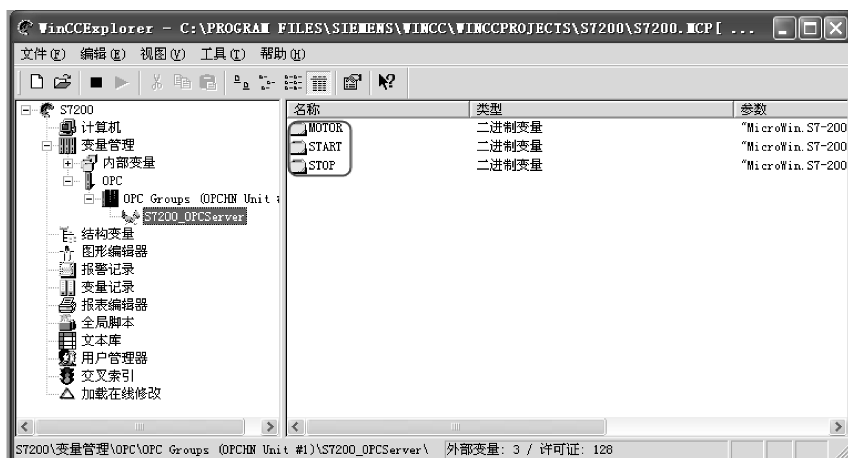


图 7-18　添加变量（2）

选择“背景颜色”，用鼠标右键单击右边的灯泡图标，弹出快捷菜单，如图 7-19 所示，单击“动态对话框”，弹出“动态范围”，如图 7-20 所示，单击按钮，弹出如图 7-21 所示的界面，将触发器改为“有变化时”，将变量和“MOTOR”链接。



图 7-19　对象属性设置

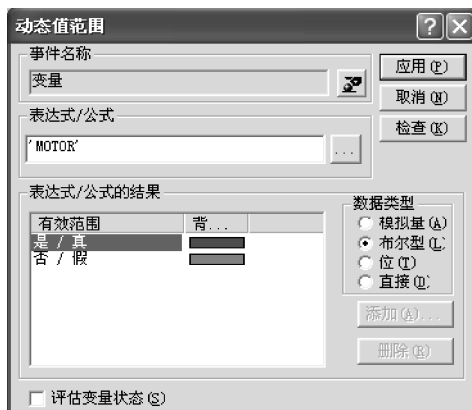


图 7-20　动态范围



图 7-21　改变触发器

将变量“START”与启动按钮连接在一起。双击画面上的“START”按钮，弹出对象属性界面，如图7-22所示，选中“事件”选项卡，再选中“鼠标”→“按左键”→(动作)，单击鼠标右键，弹出快捷菜单，单击“直接连接”，弹出直接连接界面，如图7-23所示，在“常数”中输入“1”，在“变量”中选定参数“START”，如图7-23所示，这样操作的含义是用鼠标左键单击“START”按钮时，变量“START”赋值为1。同样释放鼠标左键时，将变量“START”赋值为0，如图7-24所示。

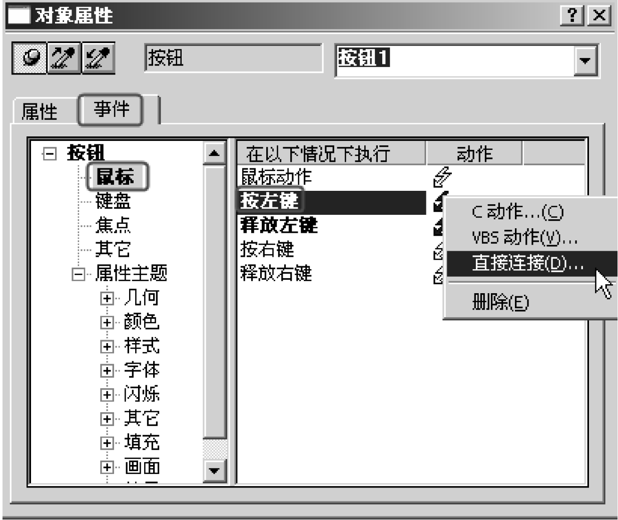


图7-22　对象属性



图7-23　直接连接（1）

将变量“STOP”与停止按钮连接在一起，方法与前述类似，在此不再赘述。

⑦ 保存工程。在图形编辑器界面中，保存工程。

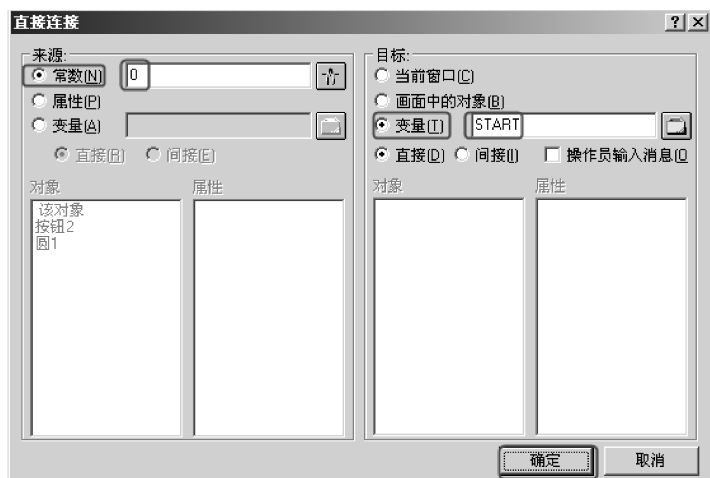


图 7-24　直接连接（2）

⑧ 运行和显示。在图形编辑器界面中，单击“激活”按钮 ，再单击“START”按钮灯为红色，单击“STOP”按钮，灯为灰色，如图 7-25 所示。

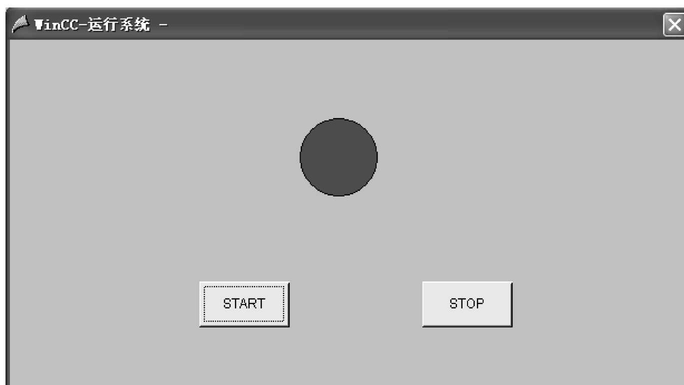


图 7-25　运行和显示

小　　结

（1）掌握建一个组态软件工程的一般方法。

（2）组态软件中的变量、动画和命令程序。

习　　题

1. 阐述组态软件的功能和发展趋势。
2. 国内外还有哪些知名的组态软件？这些知名的组态软件有何特点？
3. 用 WinCC 建立一个简单工程大致需要哪些步骤？

第 8 章　S7－200 PLC 的高速输入和高速输出功能及其应用

本章介绍 S7－200 系列 PLC 的高速输入和高速输出点。S7－200 的高速输入功能主要可用于测量高速脉冲的个数；S7－200 的高速输出点可以直接对步进电动机和伺服电动机进行运动控制，读者可以根据实际情况对程序和硬件配置进行移植。

8.1　S7－200 PLC 的高速输入及其应用

8.1.1　高速计数器的简介

S7－200 CPU 提供了多个高速计数器（HSC0～HSC5）以响应快速脉冲输入信号。高速计数器的计数速度比 PLC 的扫描速度要快得多，因此高速计数器可独立于用户程序工作，不受扫描时间的限制。用户通过相关指令，设置相应的特殊存储器控制计数器的工作。

1. 高速计数器的工作模式和输入

高速计数器有 12 种工作模式，每个计数器都有时钟、方向控制、复位启动等特定输入。对于双向计数器，两个时钟都可以运行在最高频率，高速计数器的最高计数频率取决于 CPU 的类型。在正交模式下，可选择 1×(1 倍速）或者 4×(4 倍速）输入脉冲频率的内部计数频率。高速计数器有 8 种 4 类工作模式。具体介绍如下：

1）无外部方向输入信号的加/减计数器（模式 0 和模式 2)。用高数计数器的控制字的第 3 位控制加/减计数，该位为 1 时为加计数，为 0 时为减计数。

2）有外部方向输入信号的加/减计数器（模式 3 和模式 5)。方向信号为 1 时，为加计数，方向信号为 0 时，为减计数。

3）有加计数时钟脉冲和减计数时钟脉冲输入的双相计数器（模式 6 和模式 8)。若加计数脉冲和减计数脉冲的上升沿出现的时间间隔短，高速计数器认为这两个事件同时发生，当前值不变，也不会有计数方向变化的指示。否则高速计数器能捕捉到每一个独立的信号。

4）A/B 相正交计数器（模式 9 和模式 11)。它的两路计数脉冲的相位相差 90°，正转时 A 相时钟脉冲比 B 相时钟脉冲超前 90°。反转时，A 相时钟脉冲比 B 相时钟脉冲滞后 90°。利用这一特点，正转时加计数，反转时，减计数。

高速计数器的工作模式和输入点见表 8-1。

表 8-1　高速计数器的工作模式和输入点

模　　式	中断描述	输　入　点			
	HSC0	I0.0	I0.1	I0.2	
	HSC1	I0.6	I0.7	I1.0	I1.1
	HSC2	I1.2	I1.3	I1.4	I1.5
	HSC3	I0.1			
	HSC4	I0.3	I0.4	I0.5	
	HSC5	I0.4			
0	带有内部方向控制的单相计数器	时钟			
1		时钟		复位	
2		时钟		复位	启动
3	带有外部方向控制的单相计数器	时钟	方向		
4		时钟	方向	复位	
5		时钟	方向	复位	启动
6	带有增减计数时钟的双相计数器	增时钟	减时钟		
7		增时钟	减时钟	复位	
8		增时钟	减时钟	复位	启动
9	A/B 正交计数器	时钟 A	时钟 B		
10		时钟 A	时钟 B	复位	
11		时钟 A	时钟 B	复位	启动
12	只有 HSC0 和 HSC3 支持模式 12				

【关键点】 S7-200 CPU 221、CPU 222 没有 HSC1 和 HSC2；CPU 224、CPU 224XP 和 CPU 226 拥有全部的 6 个高速计数器。只有 HSC0 和 HSC3 支持模式 12，其中 HSC0 计数 Q0.0 的输出脉冲，其中 HSC3 计数 Q0.1 的输出脉冲，在此模式下工作时，并不需要外部接线。

高速计数器的硬件输入接口与普通数字量接口使用相同的地址。已经定义用于高速计数器的输入点不能再用于其他功能。但某些模式下，没有用到的输入点还可以用作开关量输入点。

2. 高速计数器的控制字和初始值、预置值

所有的高速计数器在 S7-200 CPU 的特殊存储区中都有各自的控制字。控制字用来定义计数器的计数方式和其他一些设置，以及在用户程序中对计数器的运行进行控制。高速计数器的控制字的位地址分配见表 8-2。

表 8-2　高速计数器的控制字的位地址分配表

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	描　　述
SM37.0	SM47.0	SM57.0	—	SM147.0	—	复位有效控制，0 = 复位高电平有效，1 = 复位低电平有效
—	SM47.1	SM57.1	—	—	—	启动有效控制，0 = 启动高电平有效，1 = 启动低电平有效

（续）

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	描　　述
SM37.2	SM47.2	SM57.2	—	SM147.2	—	正交计数器速率选择，0 = 4 × 计数率，1 = 1 × 计数率
SM37.3	SM47.3	SM57.3	SM137.3	SM147.3	SM157.3	计数方向控制，0 = 减计数，1 = 加计数
SM37.4	SM47.4	SM57.4	SM137.4	SM147.4	SM157.4	向 HSC 中写入计数方向，0 = 不更新，1 = 更新
SM37.5	SM47.5	SM57.5	SM137.5	SM147.5	SM157.5	向 HSC 中写入预置值，0 = 不更新，1 = 更新
SM37.6	SM47.6	SM57.6	SM137.6	SM147.6	SM157.6	向 HSC 中写入初始值，0 = 不更新，1 = 更新
SM37.7	SM47.7	SM57.7	SM137.7	SM147.7	SM157.7	HSC 允许，0 = 禁止 HSC，1 = 允许 HSC

高速计数器都有初始值和预置值，所谓初始值就是高速计数器的起始值，而预置值就是计数器运行的目标值，当前值（当前计数值）等于预置值时，会引发一个内部中断事件，初始值、预置值和当前值都是 32 位有符号整数。必须先设置控制字以允许装入初始值和预置值，并且初始值和预置值存入特殊存储器中，然后执行 HSC 指令使新的初始值和预置值有效。装载高速计数器的初始值、预置值和当前值的寄存器与计数器的对应关系见表 8-3。

表 8-3　装载高速计数器的初始值、预置值和当前值的寄存器与计数器的对应关系表

高速计数器	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5
初始值	SMD38	SMD48	SMD58	SMD138	SMD148	SMD158
预置值	SMD42	SMD52	SMD62	SMD142	SMD152	SMD162
当前值	HC0	HC1	HC2	HC3	HC4	HC5

3. 高速计数器指令介绍

高速计数器（HSC）指令根据 HSC 特殊内存位的状态配置和控制高速计数器。高速计数器定义（HDEF）指令选择特定的高速计数器（HSCx）的操作模式。模式选择定义高速计数器的时钟、方向、起始和复位功能。高速计数指令格式见表 8-4。

表 8-4　高速计数指令格式

LAD	输入/输出	参 数 说 明	数 据 类 型
HDEF: EN ENO HSC MODE	HSC	高速计数器的号码	BYTE
	MODE	模式	BYTE
HSC: EN ENO N	N	指定高速计数器的号码	WORD

以下一个简单例子说明控制字和高速计数器指令的具体应用，程序如图 8-1 所示。

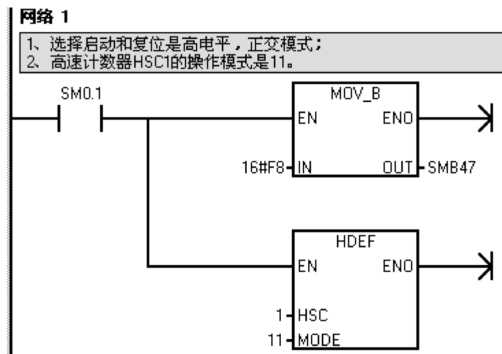
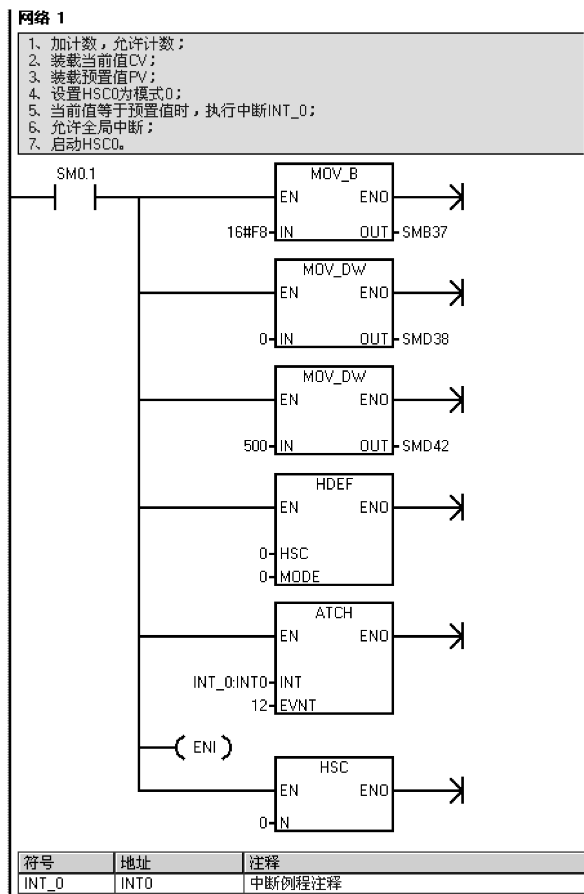


图 8-1　程序

【例 8-1】用高速计数器 HSC0 计数，当计数值达到 500 ~ 1 000 之间时报警，报警灯 Q0.0 亮。

【解】

从这个题目可以看出，报警有上位 1 000 和下位 500，因此当高速计数达到计数值时，要 2 次执行中断程序。主程序如图 8-2 所示，中断程序 0 如图 8-3 所示，中断程序 1 如图 8-4 所示。



符号	地址	注释
INT_0	INT0	中断例程注释

图 8-2　主程序

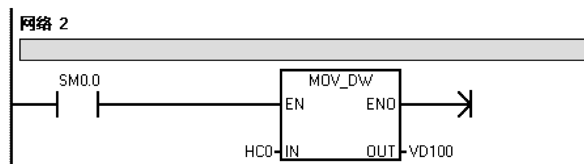


图 8-2　主程序（续）

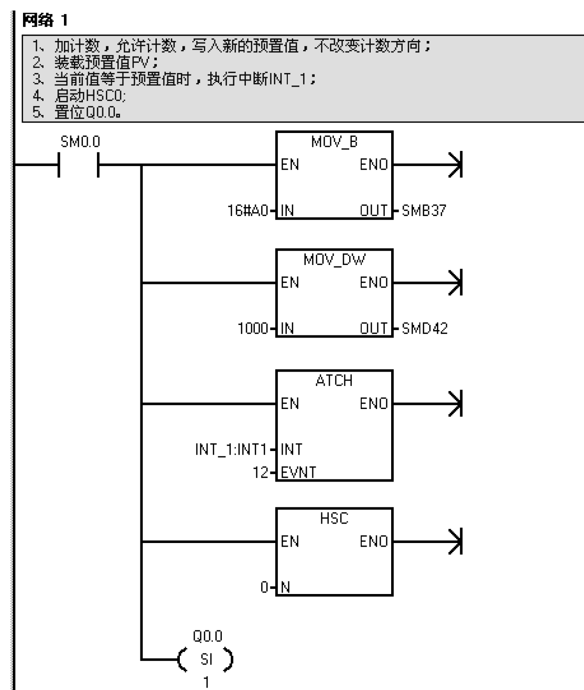


图 8-3　中断程序 0

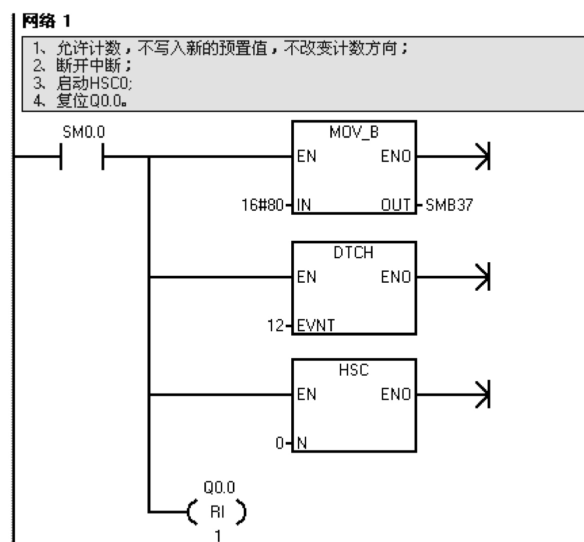


图 8-4　中断程序 1

8.1.2 高速计数器在转速测量中的应用

以下用两个例子说明高速计数器在转速测量中的应用。

【例 8-2】 一台电动机上配有一台光电编码器（光电编码器与电动机同轴安装），试用 S7－200 CPU 测量电动机的转速。

【解】

由于光电编码器与电动机同轴安装，所以光电编码器的转速就是电动机的转速。接线图如图 8-5 所示。

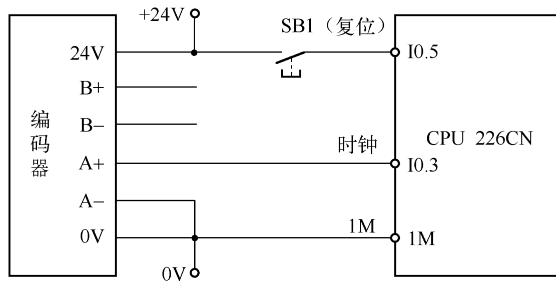


图 8-5 接线图

1. 软硬件配置

① 1 套 STEP 7－Micro/WIN V4.0 SP9。

② 1 台 CPU 226CN。

③ 1 台光电编码器（1024 线）。

④ 1 根编程电缆（或者 CP 5611 卡）。

【关键点】 光电编码器的输出脉冲信号有＋5 V 和＋24 V（或者＋18 V），而多数 S7－200 CPU 的输入端的有效信号是＋24 V（PNP 接法时），只有 CPU 224XP 型的 I0.3、I0.4 和 I0.5 三个输入端子既可以接入＋5 V 的信号，也可以接入＋24 V 的信号。因此，在选用光电编码器时要注意最好不要选用＋5 V 输出的光电编码器。图 8-5 中的编码器是 PNP 型输出，这一点也非常重要，涉及程序的初始化，在选型时要注意。此外，编码器的 A－端子要与 PLC 的 1M 短接。否则不能形成回路。

那么若只有＋5 V 输出的光电编码器是否可以直接用于以上回路测量转速呢？答案是不能，但经过晶体管升压后是可行的，具体解决方案读者自行思考。

2. 编写程序

本例的编程思路是先对高速计数器进行初始化，启动高数计数器，在 100 ms 内高速计数器计数个数，转化成每分钟编码器旋转的圈数就是光电编码器的转速，也就是电动机的转速。光电编码器为 1024 线，也就是说，高数计数器每收到 1024 个脉冲，电动机就转 1 圈。电动机的转速公式如下：

$$n=\frac{N\times 10\times 60}{1024}=\frac{N\times 75}{2^7}$$

以上公式中：n 为电动机的转速，N 为 100 ms 内高速计数器计数个数（收到脉冲个数）。

程序如图 8-6 ~ 图 8-9 所示。

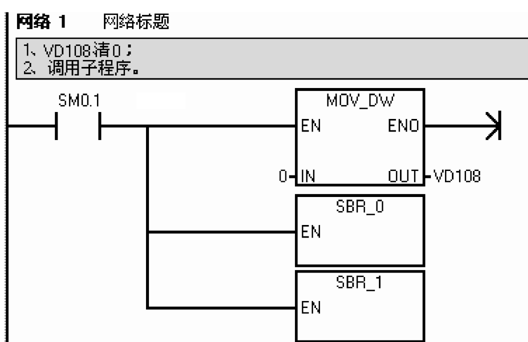


图 8-6 主程序

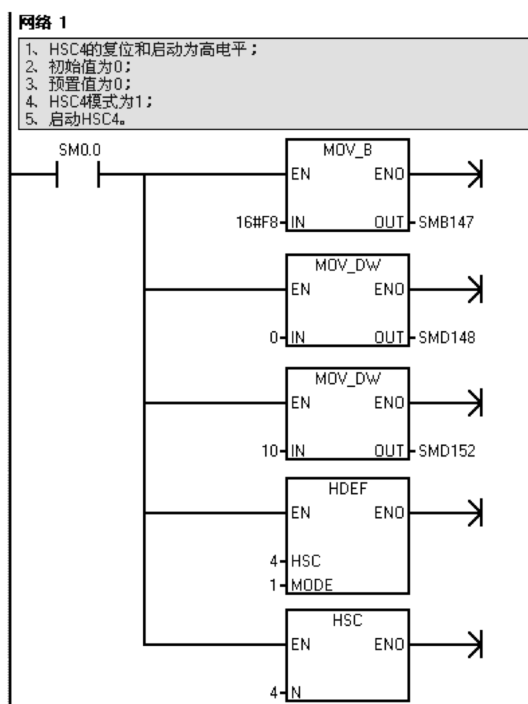


图 8-7 子程序 SBR_0

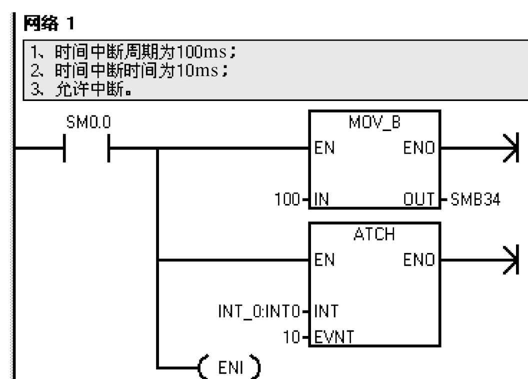


图 8-8 子程序 SBR_1

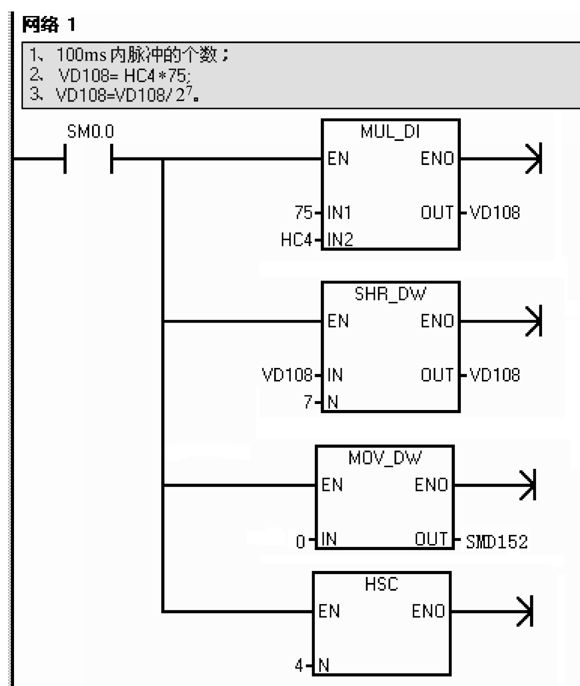


图 8-9　中断程序 INT_0

8.2　S7-200 PLC 的高速输出及其应用

8.2.1　S7-200 PLC 的高速输出简介

1. 高速脉冲输出指令介绍

高速脉冲输出功能即在 PLC 的指定输出点上实现脉冲输出（PTO）和脉宽调制（PWM）功能。S7-200 系列 PLC 配有两个 PTO/PWM 发生器，它们可以产生一个高速脉冲串或者一个脉冲调制波形。一个发生器输出点是 Q0.0，另一个发生器输出点是 Q0.1。当 Q0.0 和 Q0.1 作为高速输出点时，其普通输出点功能被禁用，而当不作为 PTO/PWM 发生器时，Q0.0 和 Q0.1 可作为普通输出点使用。一般情况下，PTO/PWM 输出负载至少为额定负载的 10%。

脉冲输出指令（PLS）配合特殊存储器用于配置高速输出功能，PLS 指令格式见表 8-5。

表 8-5　PLS 指令格式

LAD	说　明	数据类型
PLS EN　ENO Q0.X	Q0.X：脉冲输出范围，为 0 时 Q0.0 输出，为 1 时 Q0.1 输出	WORD

脉冲串操作（PTO）按照给定的脉冲个数和周期输出一串方波（占空比 50%，如图 8-10所示)。PTO 可以产生单段脉冲串或者多段脉冲串（使用脉冲包络)。可以 μs 或 ms 为单位指定脉冲宽度和周期。

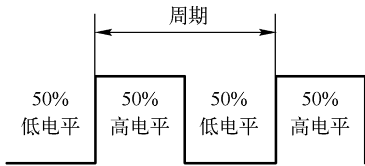


图 8-10　脉冲串输出

PTO 脉冲个数范围为 1 ~ 4 294 967 295，周期为 10 μs ~ 65 535 μs 或者 2 ms ~ 65 535 ms。

2. 与 PLS 指令相关的特殊寄存器的含义

如果要装入新的脉冲数（SMD72 或 SMD82)、脉冲宽度（SMW70 或 SMW80）和周期（SMW68 或 SMW78)，应该在执行 PLS 指令前装入这些值和控制寄存器，然后 PLS 指令会从特殊存储器 SM 中读取数据，并按照存储数值控制 PTO/PWM 发生器。这些特殊寄存器分为三大类：PTO/PWM 功能状态字、PTO/PWM 功能控制字和 PTO/PWM 功能寄存器。这些寄存器的含义见表 8-6、表 8-7 和表 8-8。

表 8-6　PTO 控制寄存器的 SM 标志

Q0. 0	Q0. 1	控 制 字 节
SM67. 0	SM77. 0	PTO/PWM 更新周期值（0 = 不更新，1 = 更新周期值）
SM67. 1	SM77. 1	PWM 更新脉冲宽度值（0 = 不更新，1 = 脉冲宽度值）
SM67. 2	SM77. 2	PTO 更新脉冲数（0 = 不更新，1 = 更新脉冲数）
SM67. 3	SM77. 3	PTO/PWM 时间基准选择（0 = 1 μs/周期，1 = 1 ms/周期）
SM67. 4	SM77. 4	PWM 更新方法（0 = 异步更新，1 = 同步更新）
SM67. 5	SM77. . 5	PTO 操作（0 = 单段操作，1 = 多段操作）
SM67. 6	SM77. 6	PTO/PWM 模式选择（0 = 选择 PTO，1 = 选择 PWM）
SM67. 7	SM77. 7	PTO/PWM 允许（0 = 禁止，1 = 允许）

表 8-7　其他 PTO/PWM 寄存器的 SM 标志

Q0. 0	Q0. 1	控 制 字 节
SMW68	SMW78	PTO/PWM 周期值（范围：2 ~ 65 535）
SMW70	SMW80	PWM 脉冲宽度值（范围：0 ~ 65 535）
SMD72	SMD82	PTO 脉冲计数值（范围：1 ~ 4 294 967 295）
SMB166	SMB176	进行中的段数（仅用在多段 PTO 操作中）
SMW168	SMW178	包络表的起始位置，用从 V0 开始的字节偏移表示（仅用在多段 PTO 操作中）
SMB170	SMB180	线性包络状态字节
SMB171	SMB181	线性包络结果寄存器
SMD172	SMD182	手动模式频率寄存器

表 8-8　PTO/PWM 控制字节参考

控制寄存器（十六进制）	允　　许	执行 PLS 指令的结果				
		模式选择	PTO 段操作	时基	脉冲数	周期
16#81	Yes	PTO	单段	1 μs/周期		装入

（续）

控制寄存器（十六进制）	允　许	执行 PLS 指令的结果				
		模式选择	PTO 段操作	时基	脉冲数	周期
16#84	Yes	PTO	单段	1 μs/周期	装入	
16#85	Yes	PTO	单段	1 μs/周期	装入	装入
16#89	Yes	PTO	单段	1 ms/周期		装入
16#8C	Yes	PTO	单段	1 ms/周期	装入	
16#A0	Yes	PTO	单段	1 ms/周期	装入	装入
16#A8	Yes	PTO	单段	1 ms/周期		

使用 PTO/PWM 功能相关的特殊存储器 SM 还有以下几点需要注意：

① 如果要装入新的脉冲数（SMD72 或 SMD82）、脉冲宽度（SMW70 或 SMW80）或者周期（SMW68 或 SMW78），应该在执行 PLS 指令前装入这些数值到控制寄存器。

② 如果要手动终止一个正在进行的 PTO 包络，要把状态字中的用户终止位（SM66.5 或者 SM76.5）置 1。

③ PTO 状态字中的空闲位（SM66.7 或者 SM76.7）标志着脉冲输出完成。另外，在脉冲串输出完成时，可以执行一段中断服务程序。如果使用多段操作时，可以在整个包络表完成后执行中断服务程序。

8.2.2　PWM 的应用

以下用一个例子说明 PWM 的应用。

【例 8-3】 用 CPU 226CN 的 Q0.0 输出一串脉冲，周期为 100 ms，脉冲宽度时间为 20 ms，要求有启停控制。

【解】

梯形图如图 8-11 所示。

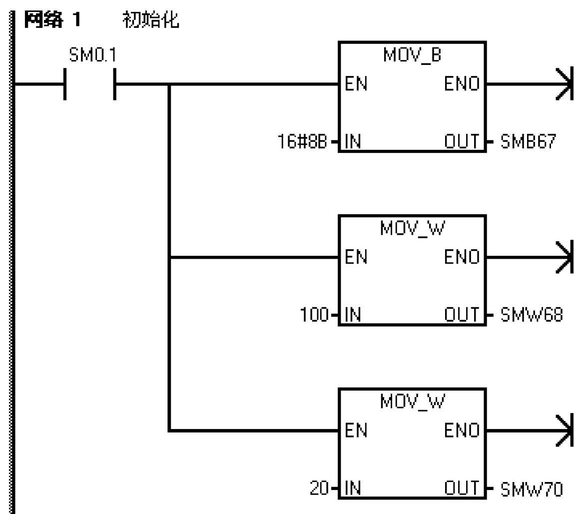


图 8-11　梯形图

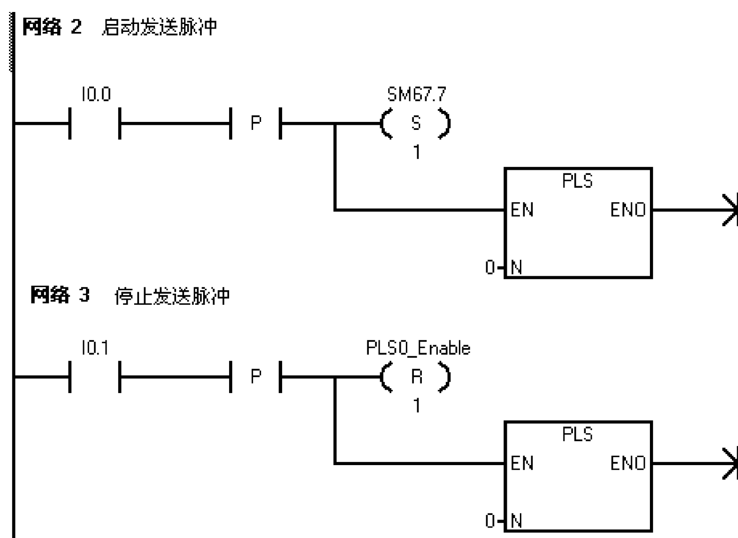


图 8-11 梯形图（续）

8.2.3 S7-200 系列 PLC 的高速输出点控制步进电动机

1. 步进电动机简介

步进电动机是一种将电脉冲转化为角位移的执行机构。一般电动机是连续旋转的，而步进电动机的转动是一步一步进行的。每输入一个电脉冲信号，步进电动机就转动一个角度。通过改变脉冲频率和数量，即可实现调速和控制转动的角位移大小，具有较高的定位精度，其最小步距角可达 0.36°，转动、停止、反转反应灵敏、可靠。在开环数控系统中得到了广泛的应用。

（1）步进电动机的分类

步进电动机可分为永磁式步进电动机、反应式步进电动机和混合式步进电动机。

（2）步进电动机的重要参数

1）步距角。它表示控制系统每发一个步进脉冲信号，电动机所转动的角度。电动机出厂时给出了一个步距角的值，这个步距角可以称之为“电动机固有步距角”，它不一定是电动机实际工作时的真正步距角，真正的步距角和驱动器有关。

2）相数。步进电动机的相数是指电动机内部的线圈组数，目前常用的有二相、三相、四相、五相等步进电动机。电动机相数不同，其步距角也不同，一般二相电动机的步距角为 0.9°/1.8°、三相的为 0.75°/1.5°、五相的为 0.36°/0.72°。在没有细分驱动器时，用户主要靠选择不同相数的步进电动机来满足自己步距角的要求。如果使用细分驱动器，则“相数”将变得没有意义，用户只需在驱动器上改变细分数，就可以改变步距角。

3）保持转矩（HOLDING TORQUE）。保持转矩是指步进电动机通电但没有转动时，定子锁住转子的力矩。它是步进电动机最重要的参数之一，通常步进电动机在低速时的力矩接近保持转矩。由于步进电动机的输出力矩随速度的增大而不断衰减，输出功率也随速度的增大而变化，所以保持转矩就成为了衡量步进电动机最重要的参数之一。比如，

当人们说2 N·m 的步进电动机，在没有特殊说明的情况下是指保持转矩为2 N·m 的步进电动机。

4）钳制转矩（DETENT TORQUE）。钳制转矩是指步进电动机没有通电的情况下，定子锁住转子的力矩。由于反应式步进电动机的转子不是永磁材料，所以它没有钳制转矩。

（3）步进电动机的主要特点

① 一般步进电动机的精度为步进角的3% ~5%，且不累积。

② 步进电动机外表允许的最高温度取决于不同电动机磁性材料的退磁点。步进电动机温度过高时，会使电动机的磁性材料退磁，从而导致力矩下降乃至于失步，因此电动机外表允许的最高温度应取决于不同电动机磁性材料的退磁点；一般来讲，磁性材料的退磁点都在130℃以上，有的甚至高达200℃以上，所以步进电动机外表温度在80 ~90℃完全正常。

③ 步进电动机的力矩会随转速的升高而下降。当步进电动机转动时，电动机各相绕组的电感将形成一个反向电动势；频率越高，反向电动势越大。在它的作用下，电动机随频率（或速度）的增大而相电流减小，从而导致力矩下降。

④ 步进电动机低速时可以正常运转，但若高于一定速度就无法起动，并伴有啸叫声。步进电动机有一个技术参数：空载起动频率。即步进电动机在空载情况下能够正常起动的脉冲频率，如果脉冲频率高于该值，电动机不能正常起动，可能发生丢步或堵转。在有负载的情况下，起动频率应更低。如果要使电动机达到高速转动，脉冲频率应该有加速过程，即起动频率较低，然后按一定加速度升到所希望的高频（电动机转速从低速升到高速）。

（4）步进电动机的细分

步进电动机的细分控制，从本质上讲是通过对步进电动机的励磁绕组中电流的控制，使步进电动机内部的合成磁场为均匀的圆形旋转磁场，从而实现步进电动机步距角的细分。

一般步进电动机的细分为1、2、4、8、16、64、128 和256 几种，通常细分数不超过256。例如当步进电动机的步距角为1.8°，那么当细分为2 时，步进电动机收到一个脉冲，只转动1.8°/2 =0.9°，可见控制精度提高了1 倍。细分数选择要合理，并非细分越大越好，要根据实际情况而定。细分数一般在步进驱动器上通过拨钮设定。

（5）步进电动机在工业控制领域的主要应用情况

步进电动机作为执行元件，是机电一体化的关键产品之一，广泛应用在各种家电产品中，例如打印机、磁盘驱动器、玩具、雨刷、机械手臂和录像机等。另外步进电动机也广泛应用于各种工业自动化系统中。由于通过控制脉冲个数可以很方便地控制步进电动机转过的角位移，且步进电动机的误差不积累，可以达到准确定位的目的。还可以通过控制频率很方便的改变步进电动机的转速和加速度，达到任意调速的目的，因此步进电动机可以广泛应用于各种开环控制系统中。

2. 步进电动机的接线

本系统选用的步进电动机是两相四线的步进电动机，其型号是17HS111，这种型号的步进电动机的引出线接线图如图8-12 所示。其含义是：步进电动机的四根引出线分别是红色、绿色、黄色和蓝色；其中红色引出线应该与步进驱动器的A 接线端子相联、绿色引出线应该与步进驱动器的$\overline{A}$ 接线端子相联、黄色引出线应该与步进驱动器的B 接线端子相联、

蓝色引出线应该与步进驱动器的 $\overline{B}$ 接线端子相联。

3. PLC 与步进电动机、步进驱动器的接线

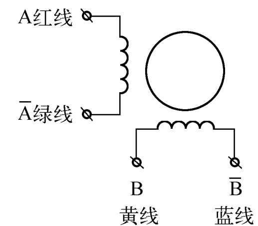


图 8-12　17HS111 型步进电动机出线接线图

步进驱动器有共阴和共阳两种接法，这与控制信号有关系，西门子 PLC 输出信号是高电平信号，所以应该采用共阴接法，顺便指出三菱的 PLC 输出的是低电平信号，因此应该采用共阳接法。

那么 PLC 能否直接与步进驱动器相连接呢？一般不能，这是因为大多数步进驱动器的控制信号是 +5 V，而西门子（以西门子 PLC 为例）PLC 的输出信号通常是 +24 V，显然是不匹配的。解决问题的办法就是在 PLC 与步进驱动器之间串联一只 2 kΩ 的电阻，起分压作用，因此输入信号近似等于 +5 V。有的资料指出串联一只 2 kΩ 的电阻是为了将输入电流控制在 10 mA左右，也就是起限流作用，在这里电阻的限流或限压作用，其含义在本质上是相同的。CP +（CP -）是脉冲接线端子，DIR +（DIR -）是方向控制信号接线端子。PLC 与步进电动机、步进驱动器的接线图如图 8-13 所示。

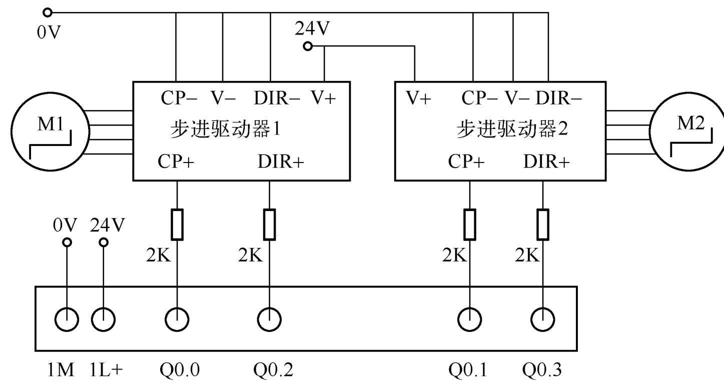


图 8-13　PLC 与步进电动机、步进驱动器的接线图

【例 8-4】 如图 8-14 所示的电气原理图，请编写梯形图实现步进电动机正转、反转和停止功能，而且正转时，反转功能失效，反之亦然。

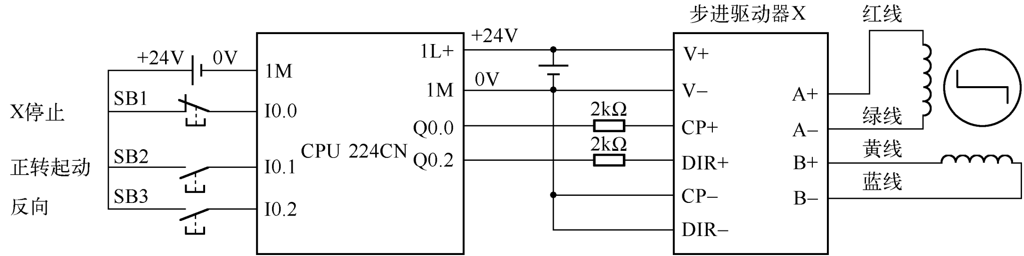
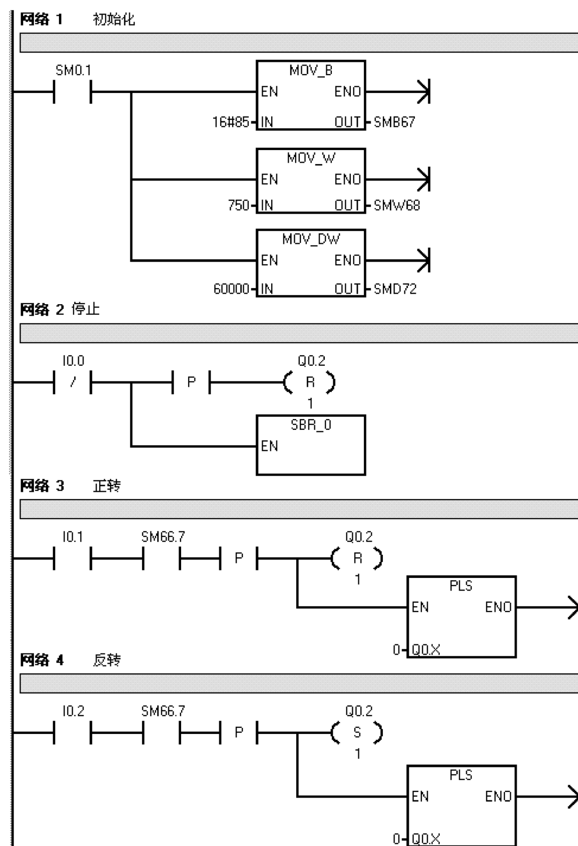


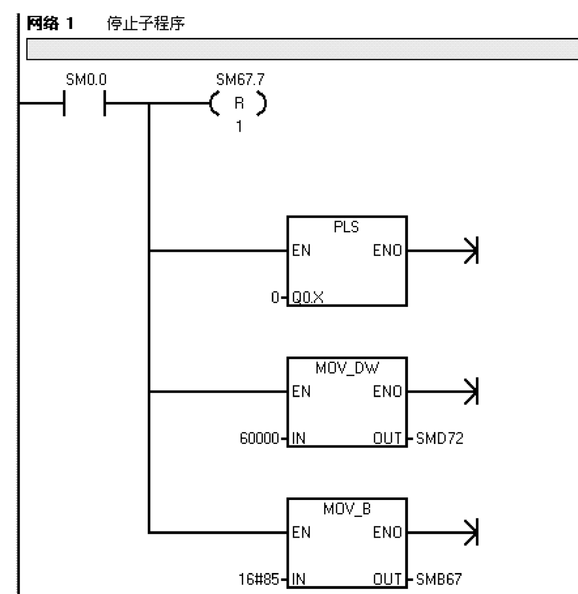
图 8-14　电气原理图

【解】

梯形图如图 8-15 所示。



a)



b)

图 8-15 程序

a）主程序 b）子程序

1. 方法一

1）主要软硬件配置：

① 1 套 STEP 7 - Micro/WIN V4.0 SP9。

② 1 台步进电动机的型号为 17HS111。

③ 1 台步进驱动器的型号为 SH - 2H042Ma。

④ 1 台 CPU 226CN。

2）步进电动机与步进驱动器的接线。本系统选用的步进电动机是两相四线的步进电动机，其型号是 17HS111，这种型号的步进电动机的出线接线图如图 8-12 所示。其含义是：步进电动机的 4 根引出线分别是红色、绿色、黄色和蓝色；其中红色引出线应该与步进驱动器的 A 接线端子相连，绿色引出线应该与步进驱动器的 $\overline{A}$ 接线端子相连，黄色引出线应该与步进驱动器的 B 接线端子相连，蓝色引出线应该与步进驱动器的 $\overline{B}$ 接线端子相联。

3）PLC 与步进电动机、步进驱动器的接线。步进驱动器有共阴和共阳两种接法，这与控制信号有关系，通常西门子 PLC 输出信号是 +24 V 信号（即 PNP 接法），所以应该采用共阴接法，所谓共阴接法就是步进驱动器的 DIR - 和 CP - 与电源的负极短接，如图 8-13 所示。

4）程序编写。

【关键点】 编写这段程序关键点在于初始化和强制使步进电动机停机而对 SMB67 的设定，其核心都在对 SMB67 寄存器的理解。其中，SMB67 = 16#85 的含义是 PTO 允许、选择 PTO 模式、单段操作、时间基准为微秒、PTO 脉冲更新和 PTO 周期更新，SMB67 = 16#CB 的含义是 PTO 禁止、选择 PTO 模式、单段操作、时间基准为微秒、PTO 脉冲不更新和 PTO 周期不更新。

若读者不想在输出端接分压电阻，那么在 PLC 的 1L + 接线端子上接 DC +5 V 也是可行的，但产生的问题是本组其他输出信号都为 DC +5 V，因此读者在设计时要综合考虑利弊，从而进行取舍。

2. 方法二

对于初学者，大多感觉利用 PLC 的高速输出点对步进电动机进行运动控制比较麻烦，特别是控制字不容易理解。幸好西门子的软件设计师早已经考虑到了这些，STEP 7 - Micro/WIN软件中提供了位置控制向导，利用位置控制向导，读者就很容易编写程序了。以下将具体介绍这种方法。

1）激活“位置控制向导”。打开 STEP 7 软件，在主菜单“工具”中选中“位置控制向导”子菜单，并单击之，弹出装置选择界面，如图 8-16 所示。

2）装置选择。S7 - 22X 系列 PLC 内部有两个装置可以配置，一个是机载 PTO/PWM 发生器，一个是 EM 253 位置模块，位置控制向导允许配置以下两个装置中的任意一个装置。很显然，本例选择“PTO/PWM 发生器”，如图 8-16 的“1”处，再单击“下一步”按钮。

3）指定一个脉冲发生器。S7 - 22X 系列 PLC 内部有两个脉冲发生器（Q0.0 和 Q0.1）可供选用，本例选用 Q0.0，再单击“下一步”按钮，如图 8-17 所示。

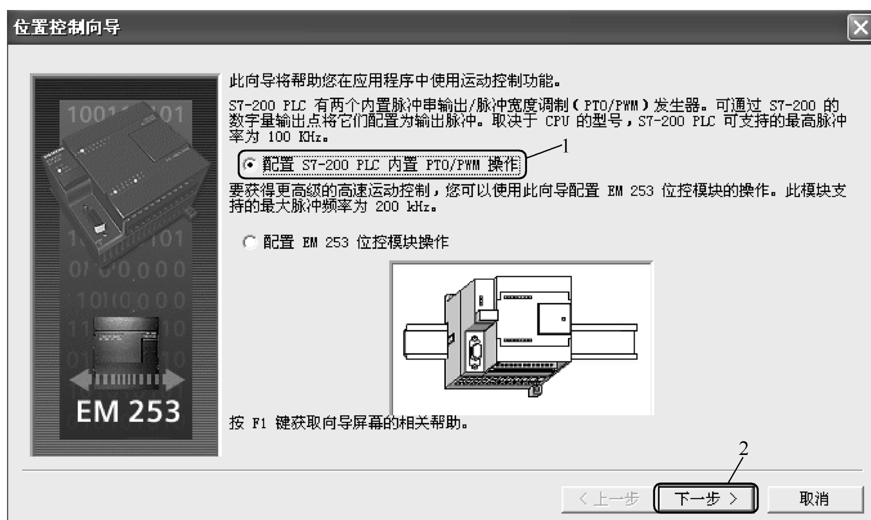


图 8-16　装置选择

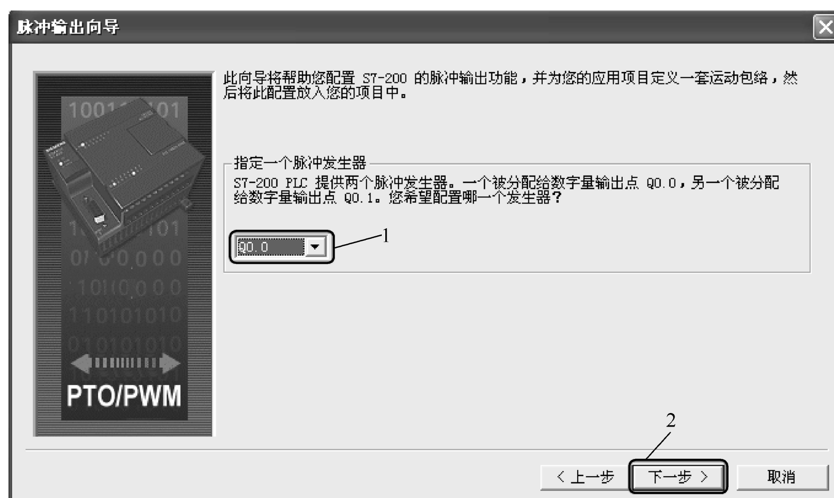


图 8-17　指定一个脉冲发生器

4）选择 PTO 或 PWM 模式，并选择时间基准。可选择 Q0.0 为脉冲串输出（PTO）或脉冲宽度调制（PWM）配置脉冲发生器，对于本例，控制步进电动机，应该选择“脉冲串输出（PTO）”，再单击“下一步”按钮，如图 8-18 所示。

5）指定电动机速度。

① MAX_SPEED：在电动机扭矩能力范围内输入应用的最佳工作速度。驱动负载所需的转矩由摩擦力、惯性和加/减速时间决定。位置控制向导会计算和显示由位控模块为指定的 MAX_SPEED 所能够控制的最低速度。

② SS_SPEED：在电动机的能力范围内输入一个数值，以低速驱动负载。如果 SS_SPEED 数值过低，电动机和负载可能会在运动开始和结束时颤动或跳动。如果 SS_SPEED 数值过高，电动机可能在起动时失步，并且在尝试停止时，负载可能使电动机不能立即停止



图 8-18　选择 PTO 或 PWM 模式

而多行走一段。

如图 8-19 所示，在“1”和“3”处输入最大速度、起动/停止速度，再单击“下一步”按钮。

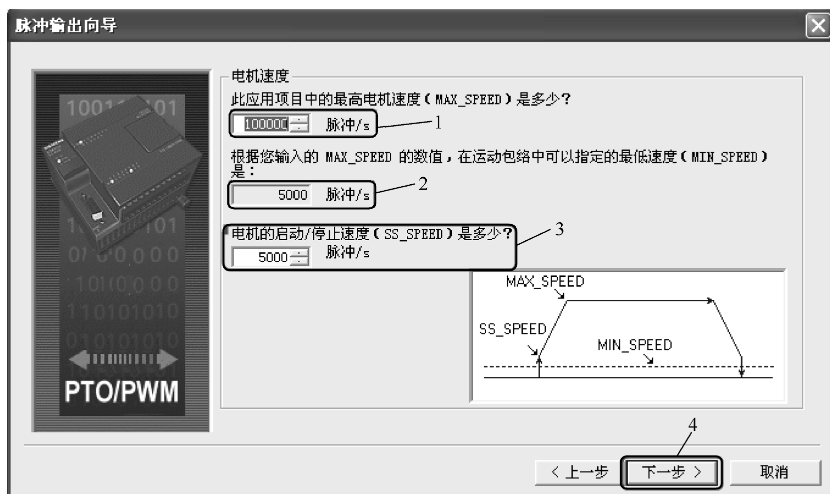


图 8-19　指定电动机速度

6）设置加速和减速时间。

① ACCEL_TIME（加速时间）：电动机从 SS_SPEED 加速至 MAX_SPEED 所需要的时间，默认值＝1000 ms（1 s），本例选默认值，如图 8-20 所示的“1”处。

② DECEL_TIME（减速时间）：电动机从 MAX_SPEED 减速至 SS_SPEED 所需要的时间，默认值＝1000 ms（1 s），本例选默认值，如图 8-20 所示的“2”处。再单击“下一步”按钮。

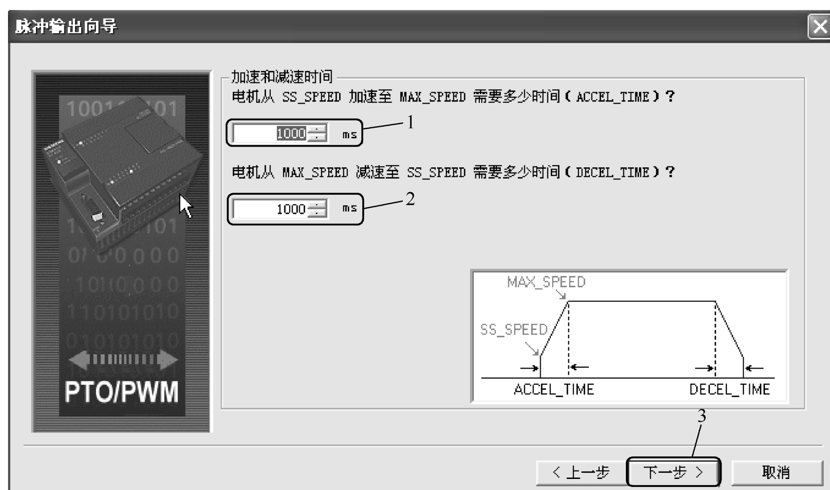


图 8-20　设置加速和减速时间

7）定义每个已配置的轮廓。先单击如图 8-21 所示中的“新包络”，弹出如图 8-22 所示界面，单击“确定”按钮，弹出如图 8-23 所示界面。

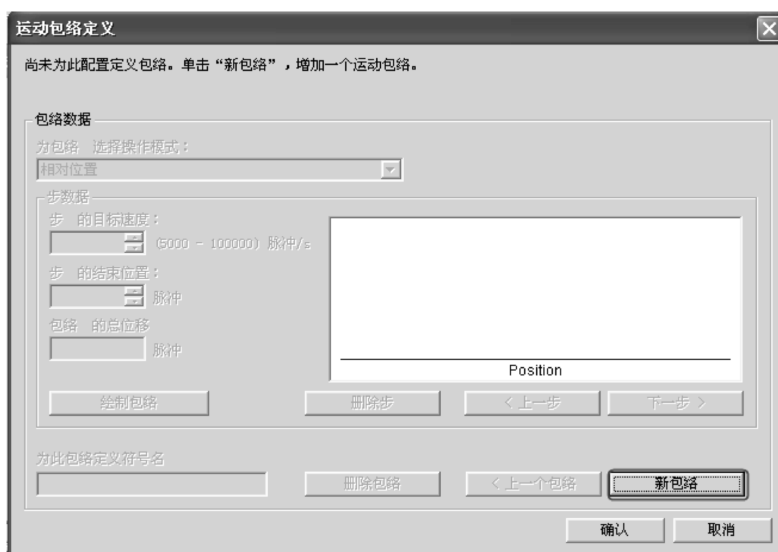


图 8-21　定义每个已配置的轮廓（1）



图 8-22　定义每个已配置的轮廓（2）

先选择“操作模式”，如图 8-23 所示的“1”处，根据操作模式（相对位置或单速连续旋转）配置。再在“2”处和“3”处输入目标速度和结束位置脉冲，接着单击“绘制包

络”按钮，包络线生成，最后单击“确认”按钮。

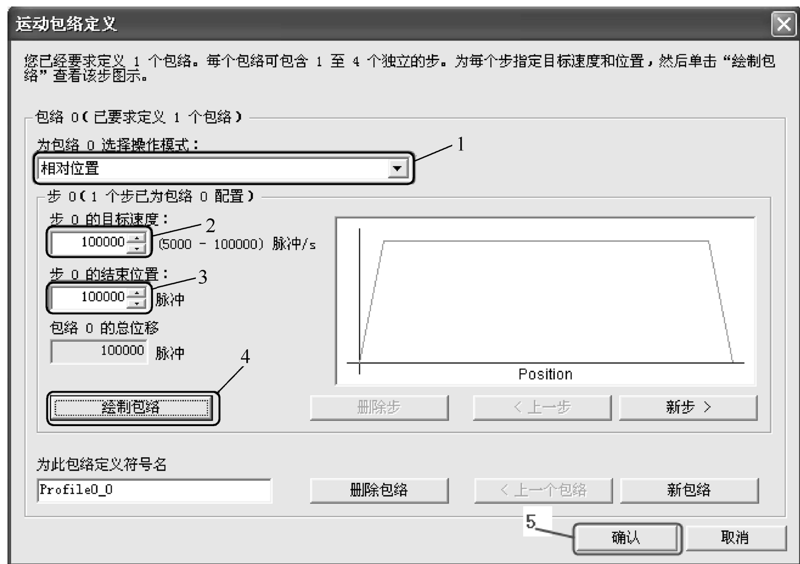


图 8-23　定义每个已配置的轮廓（3）

8）设定轮廓数据的起始 V 内存地址。PTO 向导在 V 内存中以受保护的数据块页形式生成 PTO 轮廓模板，在编写程序时不能使用 PTO 向导已经使用的地址，此地址段可以系统推荐，也可以人为分配，人为分配的好处是 PTO 向导占用的地址段可以避开读者习惯使用的地址段。设定轮廓数据的起始 V 内存地址如图 8-24 所示，本例设置为“VB1000”，再单击“下一步”按钮。



图 8-24　设定轮廓数据的起始 V 内存地址

9）生成程序代码。最后单击“完成”按钮可成生子程序，如图 8-25 所示。至此，PTO 向导的设置工作已经完成，后续工作就是在编程时使用这些生成的子程序。

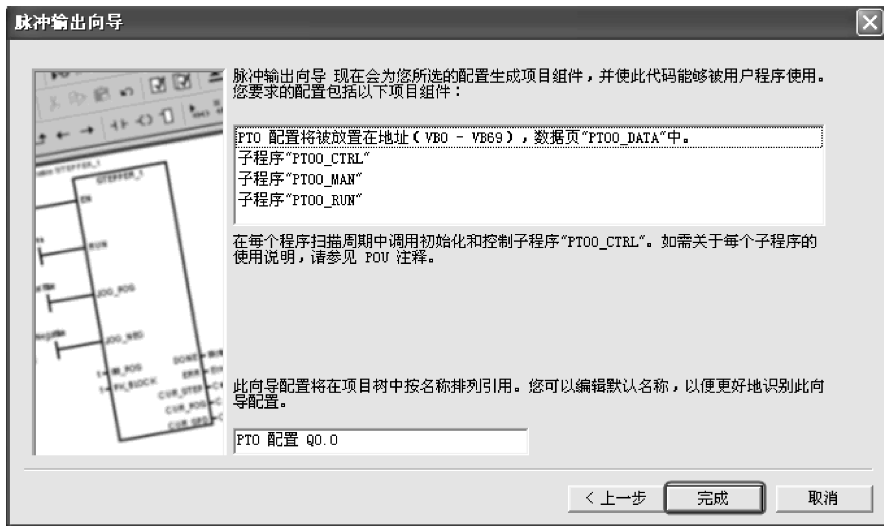


图 8-25　生成程序代码

10）子程序简介。

① PTOx_CTRL 子程序：（控制）启用和初始化与步进电动机或伺服电动机合用的 PTO 输出，在程序中只使用一次，并且确定在每次扫描时得到执行。始终用 SM0.0 作为 EN 的输入。PTOx_CTRL 子程序的参数见表 8-9。

表 8-9　PTOx_CTRL 子程序的参数表

子　程　序	各输入/输出参数的含义	数 据 类 型
PTOx_CTRL EN I_STOP D_STOP Done Error C_Pos	EN：使能	BOOL
	I_STOP：立即停止，当此输入为低电平时，PTO 功能会正常工作。当此输入变为高电平时，PTO 立即终止脉冲的发出。	BOOL
	D_STOP：表示减速停止，当此输入为低电平时，PTO 功能会正常工作。当此输入变为高电平时，PTO 会产生将电动机减速至停止的脉冲串。	BOOL
	Done：当完成任何一个子程序时，Done 参数会开启	BOOL
	C_Pos：如果 PTO 向导的 HSC 计数器功能已启用，C_ Pos 参数包含用脉冲数目表示的模块	DINT
	Error：出错时返回错误代码	BYTE

② PTOx_RUN 子程序（运行轮廓）：命令 PLC 执行存储于配置/轮廓表的特定轮廓中的运动操作。开启 EN 位会启用此子程序。PTOx_RUN 子程序的参数见表 8-10。

表 8-10　PTOx_RUN 子程序的参数表

子　程　序	各输入/输出参数的含义	数 据 类 型
PTOx_RUN：EN，START，Profile，Abort；Done，Error，C_Profile，C_Step，C_Pos	EN：使能，开启 EN 位会启用此子程序	BOOL
	START：发起“运行轮廓”的执行。对于在 START 参数已开启且 PTO 当前不活动时的每次扫描，此子程序会激活 PTO。为了确保仅发送一个命令，应以脉冲方式开启 START 参数	BOOL
	Profile：运动轮廓指定的编号或符号名	BYTE
	Abort：命令位控模块停止当前轮廓并减速至电动机停止	BOOL
	C_Profile：包含位控模块当前执行的轮廓	BYTE
	C_Step：包含目前正在执行的轮廓步骤	BYTE
	Done：当完成任何一个子程序时，Done 参数会开启	BOOL
	C_Pos：如果 PTO 向导的 HSC 计数器功能已启用，C_Pos 参数包含用脉冲数目表示的模块	DINT
	Error：出错时返回错误代码	BYTE

11）编写程序。使用了位置控制向导，编写程序就比较简单，但必须搞清楚三个子程序的使用方法，这是编写程序的关键，梯形图如图 8-26 所示。

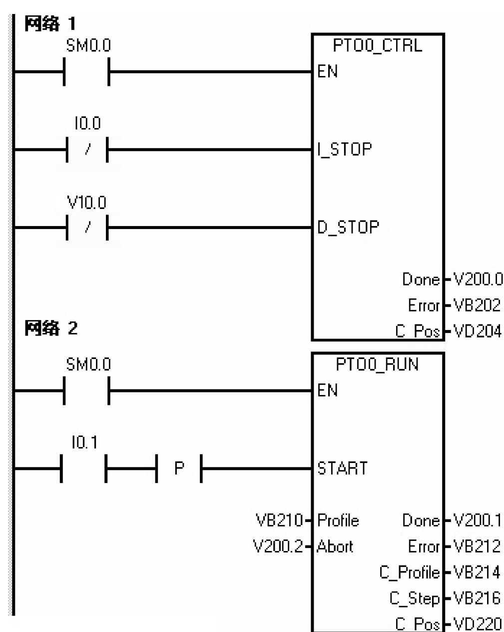


图 8-26　梯形图

【关键点】 利用指令向导编写程序，其程序简洁、容易编写，特别是控制步进电动机加速起动和减速停止，显得非常方便，且能很好避开步进电动机失步。

8.2.4　使用定位模块控制步进电动机

既然利用 S7 -200 系列的 PLC 的高速输出点可以控制步进电动机，那么为什么还要使用定位模块呢？原因在于：

① S7－200 系列的 PLC，要用其高速输出点控制步进电动机，必须具备两个条件，一是 PLC 必须要有高速输出点，二是输出点必须是晶体管输出。例如继电器输出的 PLC 即使有高速输出点也不能控制步进电动机。

② PLC 集成的高速输出点的频率不够高，以 CPU 226 为例，其最高频率只有 20 kHz，其使用范围有限，而定位模块 EM 253 的最高频率达到 200 kHz。

③ 从示波器观测波形就可以看出，定位模块高速脉冲的波形品质明显优于 PLC 内部集成的高速输出点，因此对于要求较高的控制系统必须使用定位模块。

④ PLC 集成的高速输出点的功能也不够强大。S7－200 系列的 PLC 的定位模块目前还没有发布。

下面介绍定位模块 EM 253。

相对集成在 PLC 的高速输出点，定位模块 EM 253 的功能强大，定位精度高，使用更加方便，具体特点如下：

① 用于位置开环控制回路中，不用于闭环位置模式。用于 1.1 版本以上的 CPU 22X 的扩展模块，但由于 CPU 221 自身不能带扩展模块，所以 EM 253 不能作为 CPU 221 的扩展模块。

② 可提供 12 Hz～200 kHz 的脉冲频率。

③ 支持直线和 S 曲线。

④ 提供了螺距补偿功能。

⑤ 有绝对式、手动式和相对式等多种工作模式。

⑥ 有 4 种回原点的方式。

EM 253 定位模块的外形如图 8－27 所示。其各接线端子的定义见表 8－11。

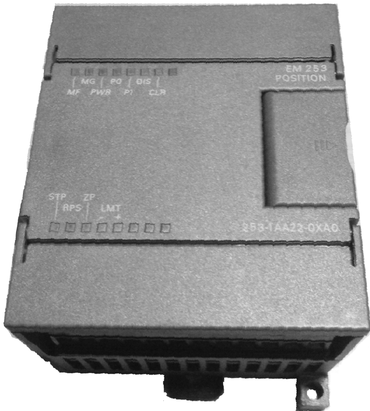


图 8－27　EM 253 模块外形图

表 8－11　接线端子的定义

序　号	接线端子	功　能	序　号	接线端子	功　能
1	L＋	DC　＋24 V	10	LMT－	负向硬限位
2	M	0 V	11	4M	
3	1M	输入 stop 信号	12	LMT＋	正向硬限位
4	STOP		11	4M	
5	2M	参考点切换输入	13	P0＋/ P0－	P0 输出（脉冲或方向）
6	RPS		14	P1＋/ P1－	P1 输出（脉冲或方向）
7	3M	零脉冲输入	15	DIS	使能控制
8	ZP		16	CLR	清除步进驱动器的脉冲计数寄存器
9	T1	与 P0、P1、DIS、CLR 联合使用			

EM 253 定位模块指示灯的状态见表 8－12。

表 8-12　EM 253 定位模块指示灯的状态

序号	输入/输出（对于 EM 253）	LED/接线端子	LED 颜色	功 能 描 述
1	—	MF	红色	检测到致命的故障时接通
2	—	MG	绿色	无故障时接通，检测到故障时以 1 Hz 闪烁
3	—	PWR	绿色	L + 和 M 接上 +24VDC 供电时接通
4	输入	STOP	绿色	输入 stop 接通时亮
5	输入	RPS	绿色	参考点切换输入接通时亮
6	输入	ZP	绿色	零脉冲输入接通时亮
7	输入	LMT -	绿色	负向限位接通时亮
8	输入	LMT +	绿色	正向限位接通时亮
9	输出	P0	绿色	P0 输出触发时亮
10	输出	P1	绿色	P1 输出触发时亮
11	输出	DIS	绿色	DIS 输出时亮
12	输出	CLR	绿色	清除偏差计数输出激活时亮

下面用一个例子来讲解 EM 253 模块的使用。

【例 8-5】某设备上有一套步进驱动系统，步进驱动器的型号为 SH - 2H042Ma，步进电动机的型号为 17HS111，是两相四线 DC 24 V 步进电动机，要求：压下按钮 SB1 时，步进电动机带动 X 方向运动，系统有两个限位开关进行硬件限位。请画出 I/O 接线图并编写程序。

【解】

1. 软硬件的配置和接线图

1）主要软硬件配置：

① 1 套 STEP 7 - Micro/WIN V4.0 SP9。

② 1 台步进电动机的型号为 17HS111。

③ 1 台步进驱动器的型号为 SH - 2H042Ma。

④ 1 台 CPU 226CN。

⑤ 1 台 EM 253 模块。

2）步进电动机与步进驱动器的接线。本系统选用的步进电动机是两相四线的步进电动机，其型号是 17HS111，这种型号的步进电动机的出线接线图如图 8-28 所示。其含义是：步进电动机的 4 根引出线分别是红色、绿色、黄色和蓝色；其中红色引出线应该与步进驱动器的 A 接线端子相连，绿色引出线应该与步进驱动器的 $\overline{A}$ 接线端子相连，黄色引出线应该与步进驱动器的 B 接线端子相连，蓝色引出线应该与步进驱动器的 $\overline{B}$ 接线端子相联。

3）EM 253 定位模块与步进驱动器的接线。步进驱动器的脉冲输入端子与定位模块的 P0 脉冲输出端子相连，步进驱动器的方向控制端子与 P1 端子相连，而步进驱动器的 V - 端子和 EM 253 的 M 端子相连，达到共地的目的。由于本驱动器没有使能端子所以 EM 253 的 DIS 端子不接线。

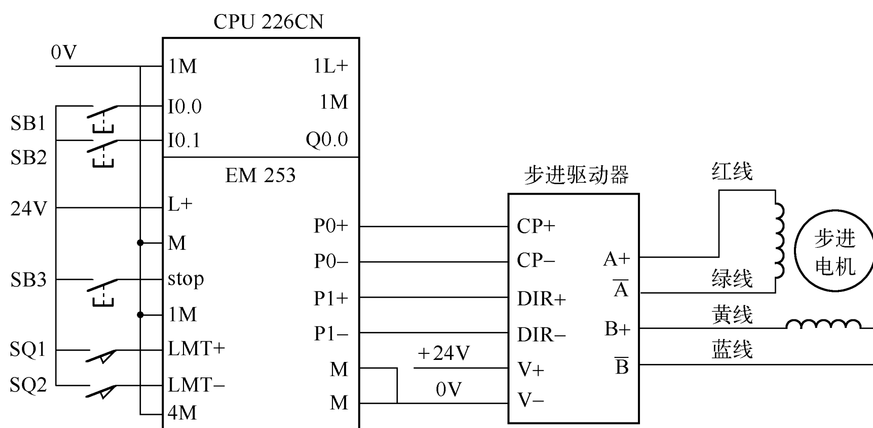


图 8-28　PLC 与驱动器和步进电动机接线图

2. 系统的硬件组态

1）打开“位置控制向导”配置工具。启动软件 STEP 7 - Micro/WIN V4.0，单击导航条中的“工具”按钮，再单击“位置控制向导”按钮，弹出“位置控制向导”界面，如图 8-29所示，也可以通过单击主菜单下的“位置控制向导”子菜单实现。

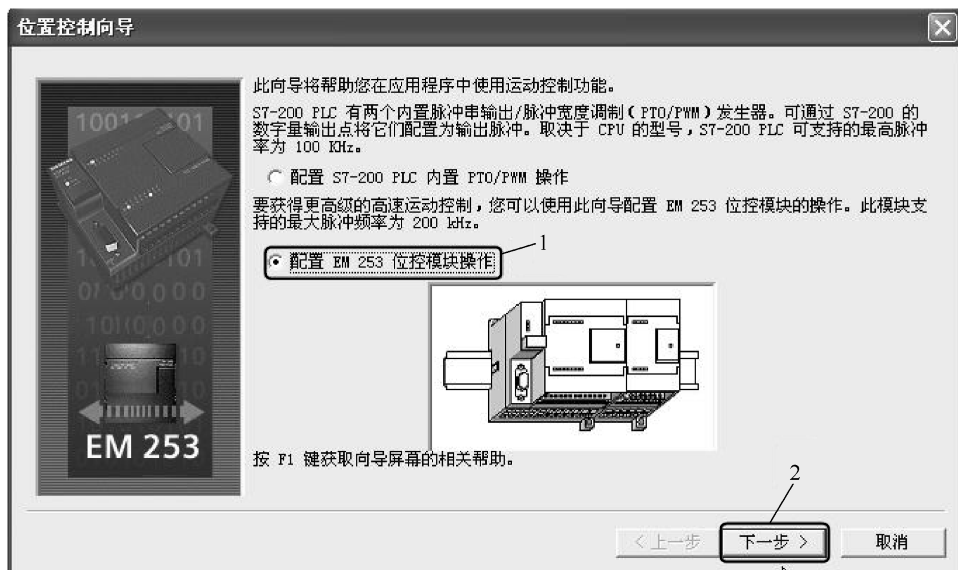


图 8-29　位置控制向导

2）选择位置配置模式。先选定“配置 EM 253 的位置模块操作”，再单击“下一步”按钮。

3）读取定位模块 EM 253 的逻辑位置。先设定模块的位置，由于只有一个扩展模块，所以模块位置为 0，再单击“读取模块”按钮，弹出位置 4 处的信息，再单击“下一步”按钮，读取定位模块 EM 253 的逻辑位置如图 8-30 所示。

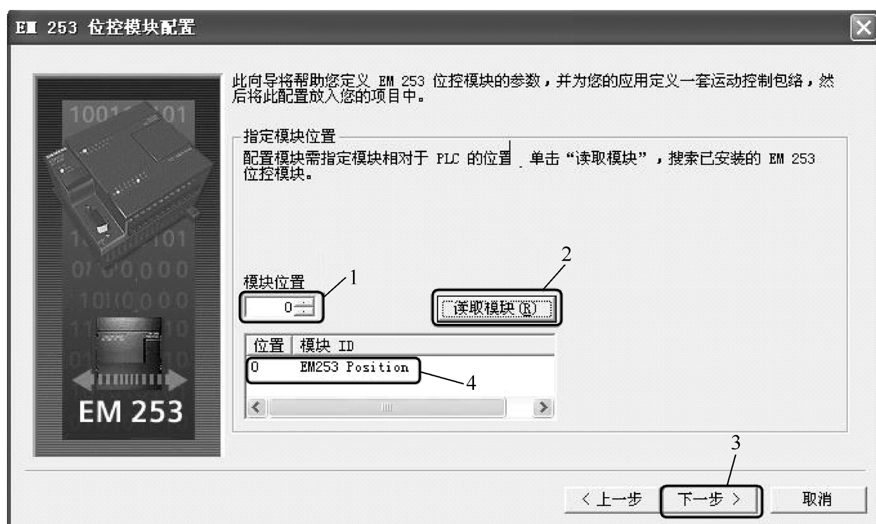


图 8-30　读取定位模块 EM 253 的逻辑位置

4）输入系统的测量单位。先在标记1 处输入电动机转一周应输出的脉冲数，再在标记2 处输入基准测量单位，本例为“mm”，再在标记 3 处输入电动机旋转一周产生多少“mm”的运动，这个数值与机械结构有关，最后单击“下一步”按钮，如图 8-31 所示。

5）定义模块输入信号 LMT +、LMT 和 STP 的功能。本例选择的都是“减速停止”，当然也可根据实际需要选择“立即停止”或者“不停止”选项，如图 8-32 所示。

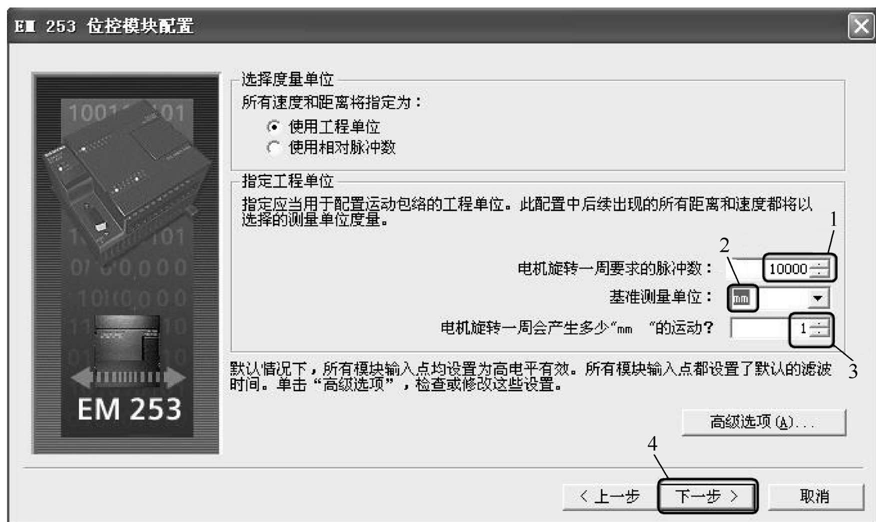


图 8-31　输入系统的测量单位

6）定义电动机速度。根据需要定义电动机的最大速度、最低速度以及起动/停止速度，再单击“下一步”按钮，如图 8-33 所示。



图 8-32　定义模块输入信号 LMT + 、LMT 和 STP 的功能

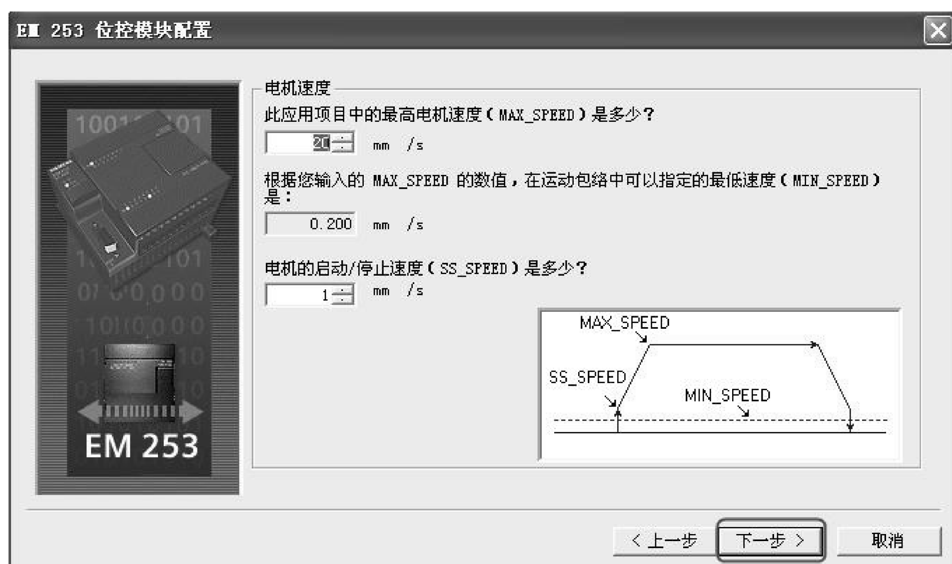


图 8-33　定义电动机速度

7）定义手动操作的参数。设定点动时电动机的速度，再单击“下一步”按钮，如图 8-34所示。

8）加/减速参数设定。加速时间就是从最低速度加速到最大速度所用的时间，设置在标记 1 处，减速时间就是从最高速度加速到最低速度所用的时间，设置在标记 2 处，再单击“下一步”按钮，如图 8-35所示。

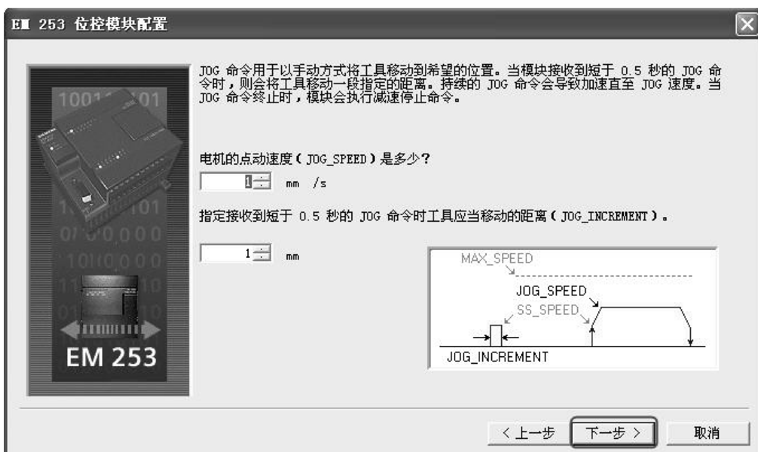


图 8-34　定义手动操作的参数

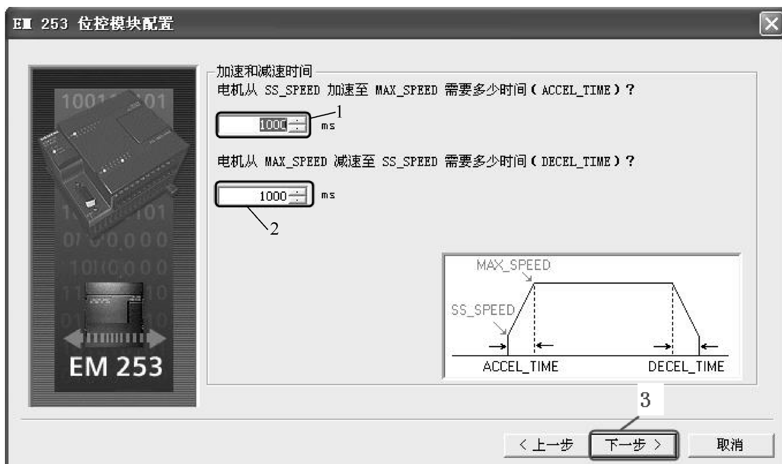


图 8-35　加/减速参数设定

9）设置运动位置拐点参数。指定补偿时间，若不补偿则输入 0，再单击“下一步”按钮，如图 8-36 所示。

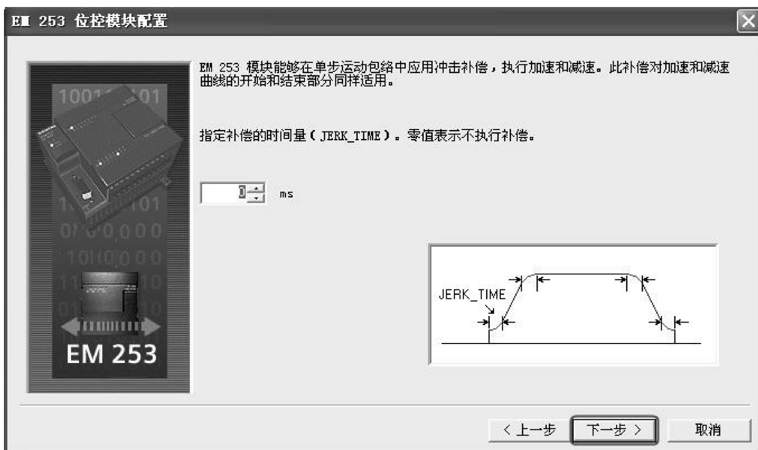


图 8-36　设置运动位置拐点参数

10）设置模块寻找原点位置参数。先选取标记1处的选项，再单击“下一步”按钮，如图8-37、图8-38和图8-39所示。



图8-37　设置模块寻找原点位置参数（1）



图8-38　设置模块寻找原点位置参数（2）

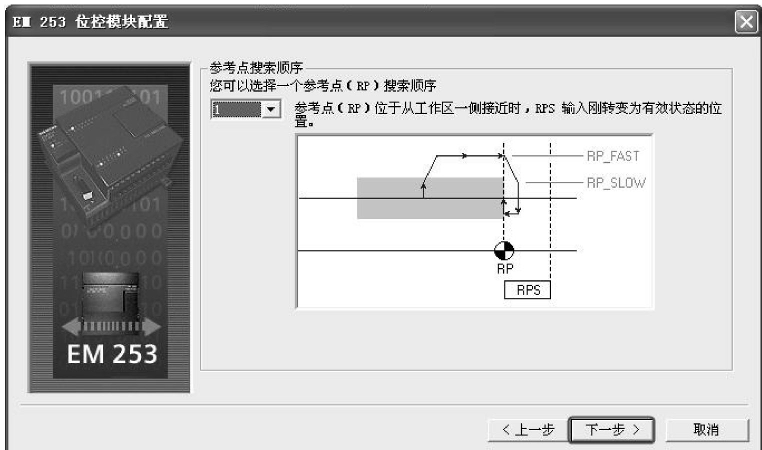


图8-39　设置模块寻找原点位置参数（3）

11）设定定位模块 EM 253 的运动轨迹包络。定位模块 EM 253 的运动轨迹包络设定如图 8-40 所示。



图 8-40　设定定位模块 EM 253 的运动轨迹包络

12）分配地址。先单击“建议地址”按钮，再单击“下一步”按钮，如图 8-41 所示。



图 8-41　分配地址

13）完成组态。单击“完成”按钮，结束组态，如图 8-42 所示。

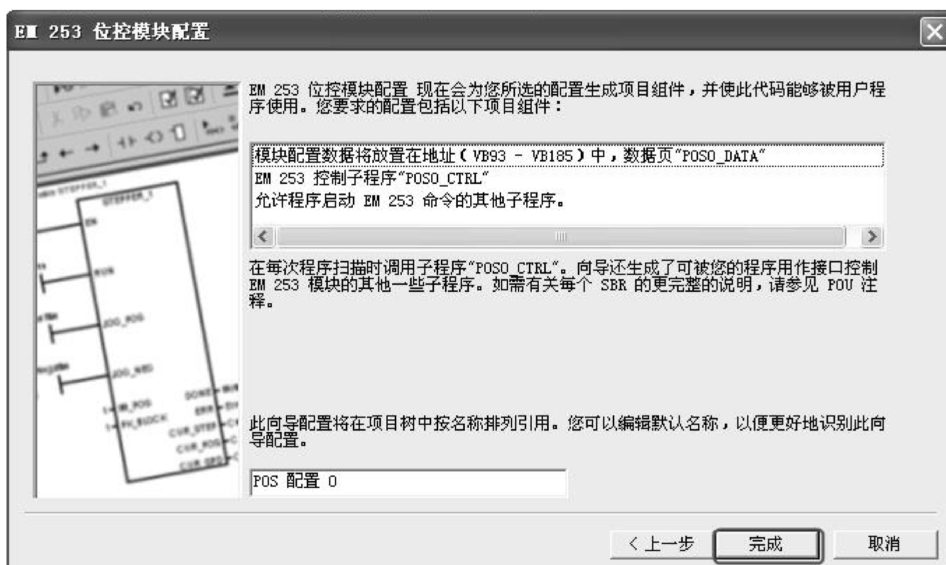


图 8-42　完成组态

3. 编写程序

1）子程序简介。完成以上的配置后，STEP 7 - Micro/WIN V4.0 SP9 自动生成一系列子程序，以下重点介绍两个，即 POSx_CTRL 和 POSx_GOTO。

① POSx_CTRL：子程序（控制）启用和初始化位控模块，方法是自动命令位控模块每次 S7 - 200 更改为 RUN 模式时，载入配置/轮廓表。在项目中仅限使用一次本子程序，并确保程序在每次扫描时呼叫本子程序。要将 SM0.0（始终开启）用作 EN 参数的输入。MOD_EN 参数必须开启，才能启用其他位置子程序向位控模块发送命令。如果 MOD_EN 参数关闭，位控模块会异常中止所有正在执行的命令。POSx_CTRL 子程序的输出参数提供位控模块的当前状态。当位控模块完成任何一个子程序时，Done 参数会开启。Error 参数包含本子程序的结果。该子程序各输入/输出参数含义见表 8-13。

表 8-13　POSx_CTRL 子程序各输入/输出参数表

子　程　序	各输入/输出参数的含义	数 据 类 型
POS0_CTRL EN Done Error MOD_~　C_Pos C_Spe~ C_Dir	EN：使能	BOOL
	MOD_EN：MOD_EN 参数必须开启，才能启用其他位置子程序向位控模块发送命令	BOOL
	C_Dir：表示电动机的当前方向	BOOL
	C_Speed：提供模块的当前速度	DINT、REAL
	C_Pos：模块的当前位置	DINT、REAL
	Done：当位控模块完成任何一个子程序时，Done 参数会开启	BOOL
	Error：出错时返回错误代码	BYTE

② POSx_GOTO：子程序命令位控模块进入所需的位置，开启 EN 位会启用此子程序。在“完成”位发出子程序执行已经完成的信号前，应确定 EN 位保持开启。开启 START 参数向位控模块发出一个 GOTO 命令。对于在 START 参数开启，且位控模块当前不繁忙时执行的每次扫描，该子程序向位控模块发送一个 GOTO 命令。为了确保仅发送了一个 GOTO 命令，应使用边缘探测元素用脉冲方式开启 START 参数。Pos 参数包含一个数值，指示移动的位置（绝对移动）或移动的距离（相对移动）。根据所选的测量单位，该数值是脉冲（DINT）或工程单位（REAL）数目。Speed 参数确定该移动的最高速度。其各输入/输出参数见表 8-14。

表 8-14　POSx_GOTO 子程序各输入/输出参数表

子　程　序	各输入/输出参数的含义	数 据 类 型
POS0_GOTO EN START Pos　Done Speed　Error Mode　C_Pos Abort　C_Spe~	EN：使能	BOOL
	START：开启 START 参数向位控模块发出一个 GOTO 命令	BOOL
	Pos：指示移动的位置（绝对移动）或移动的距离（相对移动）	BOOL
	C_Speed：提供模块的当前速度	DINT、REAL
	C_Pos：模块的当前位置	DINT、REAL
	Done：当位控模块完成本子程序时，Done 参数开启	BOOL
	Mode：移动的类型。0 - 绝对位置，1 - 相对位置，2 - 单速连续正向旋转，3 - 单速连续负向旋转	BYTE
	Error：出错时返回错误代码	BYTE

2）程序编写。编写如图 8-43 所示的程序，并将其下载到 PLC，并运行程序，当合上 SB1 时，步进电动机运行，当合上 SB2 或者 SB3 时，步进电动机停止，当碰到硬限位开关 SQ1 或者 SQ2 时，步进电动机也停止运行。

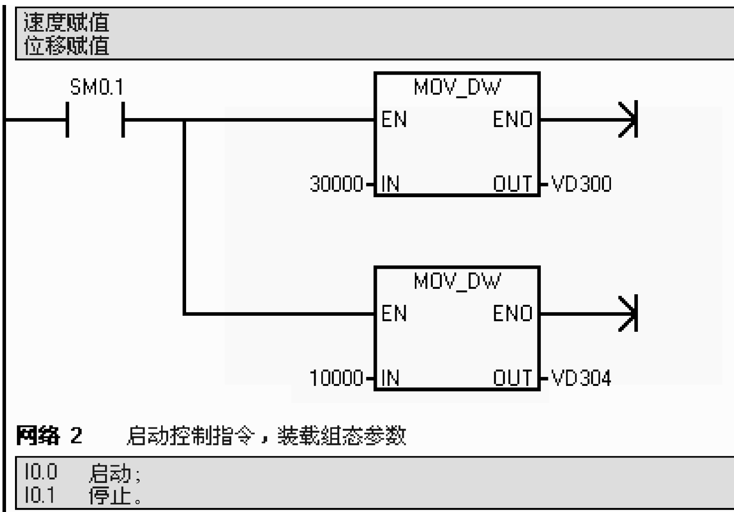


图 8-43　程序

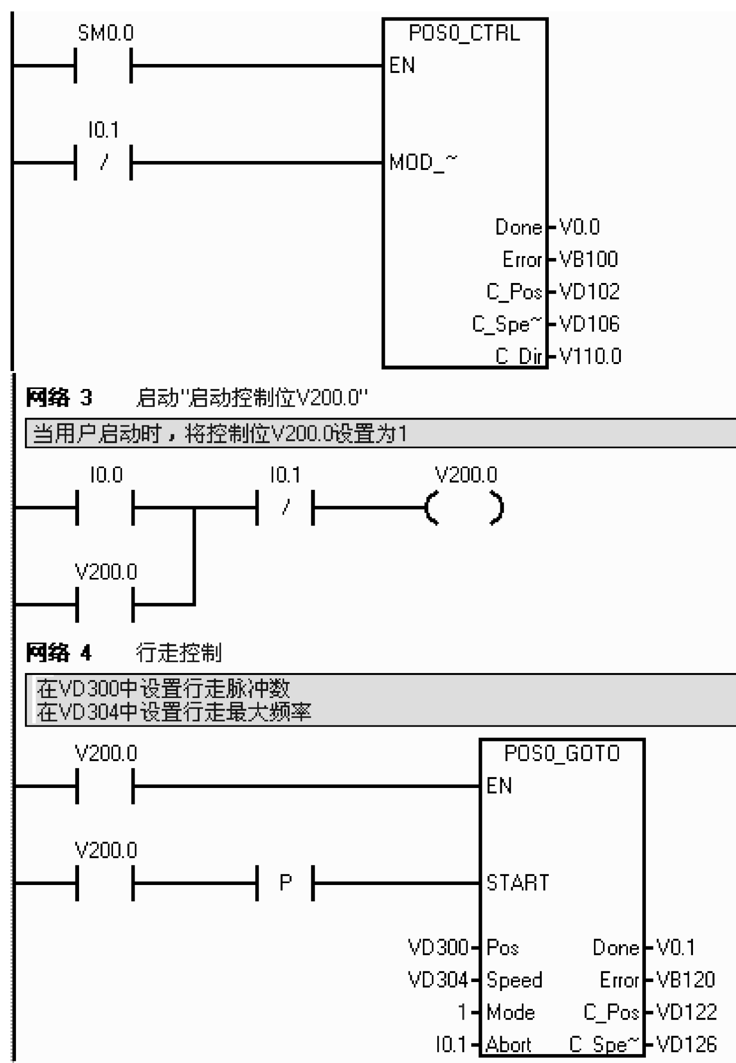


图 8-43　程序（续）

【关键点】 首先硬件系统的接线要正确，再者要注意组态正确。EM 253 用于开环控制，含义是 EM 253 不能接受反馈信号（如光电编码器的信号），并非指整个控制系统就是开环，若 EM 253 与伺服系统连接时，伺服驱动器与伺服电动机构成自闭环系统，也可以说这个控制系统是闭环的。

8.2.5　步进电动机的调速控制

1. 步进电动机的调速原理

步进电动的速度正比于脉冲频率，反比于脉冲周期，增加脉冲周期，步进电动机的速度下降，反之减小脉冲周期步进电动机的速度增加。

2. 用西门子 S7－200 控制步进电动机的调速

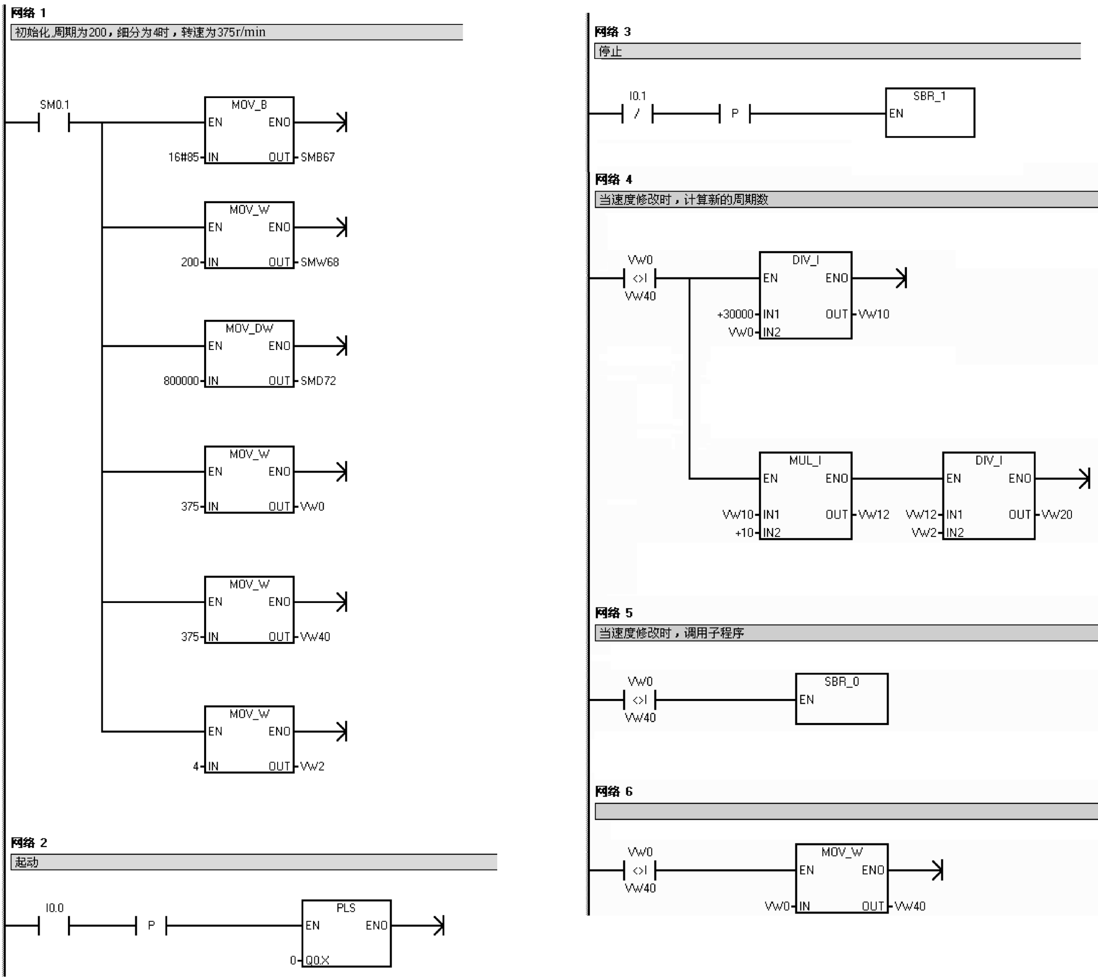
对于 S7－200 系列 PLC，脉冲周期存在特殊寄存器 SMW 68 中，因此要改变步进电动机

的转速，必须改变 SMW 68 中的脉冲频率。那么是不是只要改变了 SMW 68 中的脉冲频率，步进电动机的转速就会随之改变呢？当然不是，因为步进电动机的转速改变，除了改变 SMW 68 中的脉冲频率外，还必须是 PLC 把所有的脉冲发送完成才可以改变。因此，为了使步进电动机的转速立即改变，在改变 SMW 68 中的脉冲频率之前，必须先将步进电动机停止，这是至关重要的。以下用一个例子讲解步进电动机的调速。

【例 8-6】 已知步进电动机的步距角是 1.8°，默认情况细分为 4，默认转速为 375 r/min，转速的设定在触摸屏中进行，驱动器的细分修改后，触摸屏中的 VW2 也要随之修改。

【解】

本例的默认转速 375 r/min 存放在 VW0 中，细分数存放在 VW2 中，新的转速存放在 VW40 中。细分为 4，则步进电动机的转速实际降低到原来的四分之一。程序如图 8-44 所示。



a)

图 8-44　程序

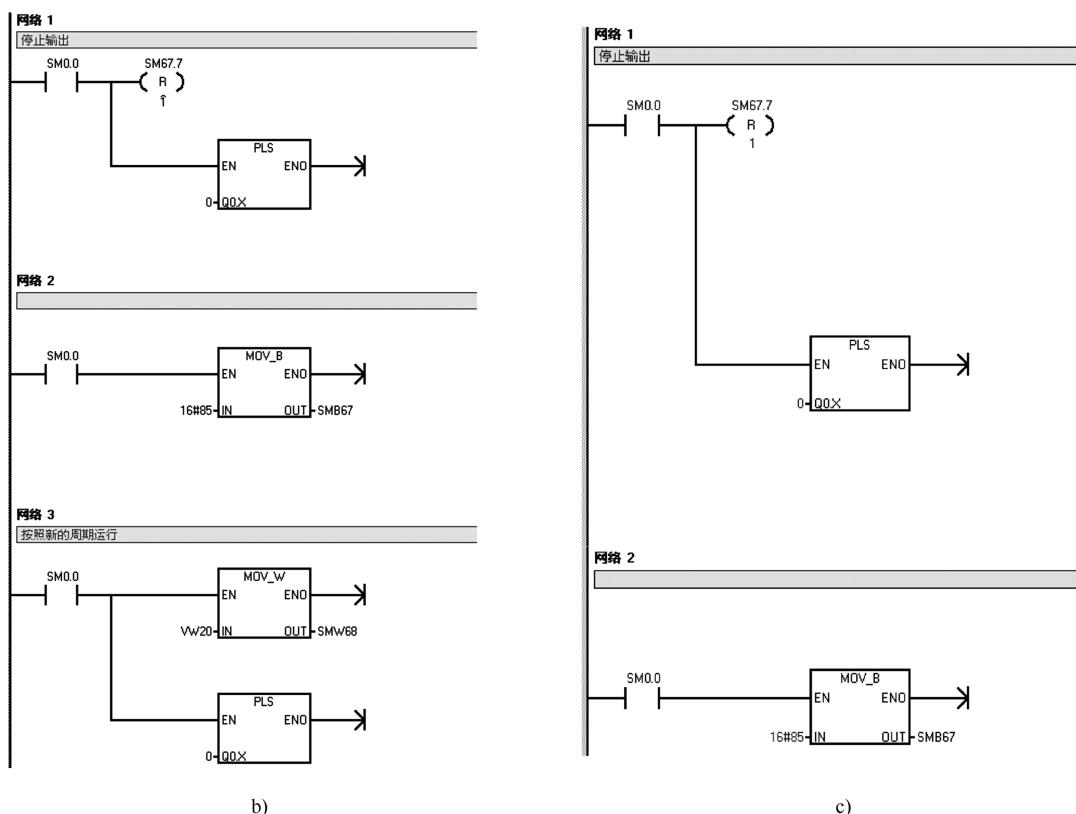


图 8-44　程序（续）

a）主程序　b）子程序 - SBR_0　c）子程序 - SBR_1

8.2.6　步进电动机的正反转控制

1. 步进电动机的正反转的原理

如图 8-14 所示，当 Q0.2 为高电平时步进电动机反转，当 Q0.2 为低电平时步进电动机正转，深层原理在此不做探讨。

2. 用西门子 S7 -200 控制步进电动机实现自动正反转

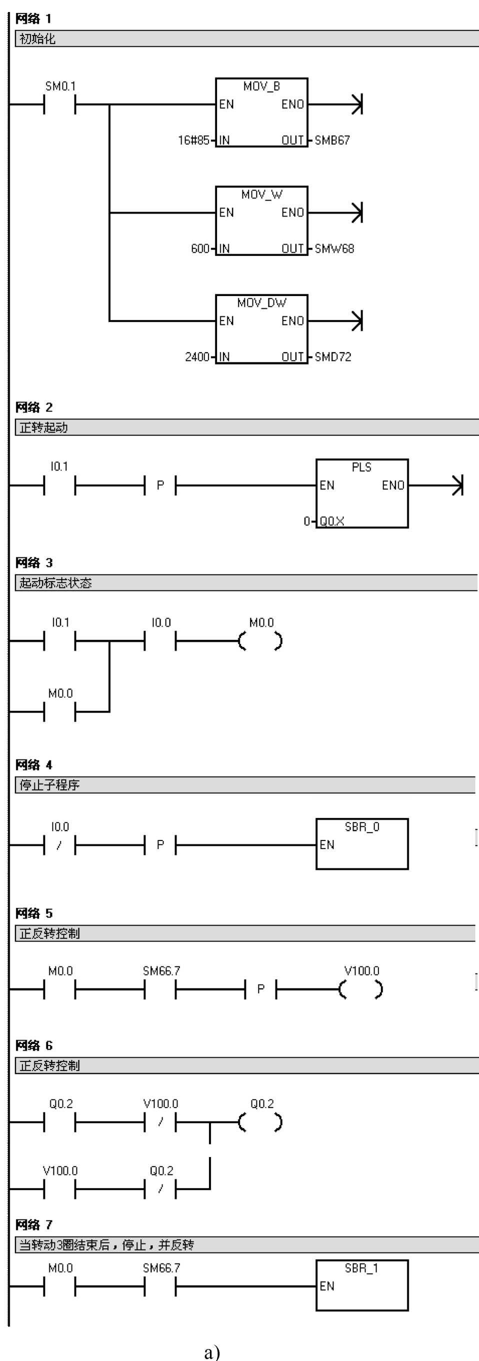
如果用按钮或者限位开关等控制步进电动机的正反转当然是很容易的在前面已经讲解过，但如果要求步进电动机自动实现正反转就比较麻烦了，下面用一个实例讲解。

【例 8-7】 已知步进电动机的步距角是 1.8°，转速为 500 r/min，要求步进电动机正转 3 圈后，再反转 3 圈，如此往复，请编写程序。

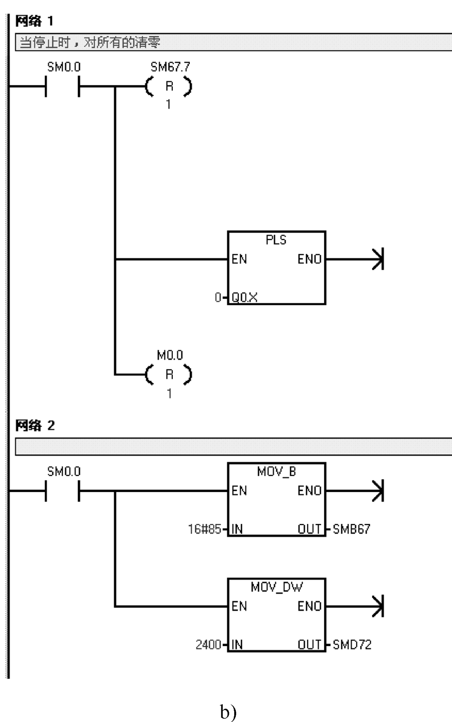
【解】

$$T = 10^6 \times \frac{60}{500 \times \frac{360°}{1.8°}} \mu s = 600\ \mu s$$，所以设定 SMW68 为 600。

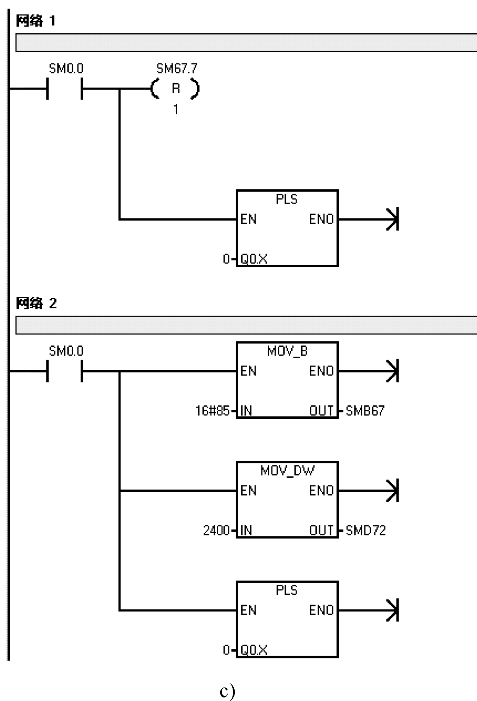
程序如图 8-45 所示。



a)



b)



c)

图 8-45　程序

a）主程序　b）子程序 - SBR_0　c）子程序 - SBR_1

8.2.7　PLC 控制伺服系统

“伺服系统”源于英文“Servomechanism System”，指经由闭环控制方式控制一个机械系统的位置、速度和加速度。伺服的概念可以从控制层面去理解，伺服的任务就是要求执行机构快速、平滑、精确地执行上位控制装置的指令要求。

三菱公司是较早研究和生产交流伺服电动机的企业之一，三菱公司早在 20 世纪 70 年代就开始研发和生产变频器，积累了较为丰富的经验，是目前少数几家能生产稳定可靠 15kW 以上伺服系统的厂家。三菱公司的伺服系统是日系产品的典型代表，它具有可靠性好、转速高、容量大、相对容易使用等特点，而且还是生产 PLC 的著名的厂家，因此其伺服系统与 PLC 产品能较好地兼容。在加工设备、自动生产线、印刷机械、纺织机械和包装机械等行业得到了广泛的应用。以下用三菱伺服系统为例进行介绍。

1. 控制系统的接线

伺服系统选用的是三菱 MR 系列。伺服电动机和伺服驱动器的连线比较简单，伺服电动机后面的编码器与伺服驱动器的连线是由三菱公司提供的专用电缆，伺服驱动器端的接口是 CN2，这根电缆一般不会接错。伺服电动机上的电源线对应连接到伺服驱动器上的接线端子上，三菱 MR－E－A 伺服系统的硬件连接示意图如图 8-46 所示，图中断路器和接触器是通用器件，只要符合要求的产品即可，电抗器和制动电阻可以根据需要选用。

接线图如图 8-47 所示。本伺服驱动器的供电电源可以是三相交流 220 V，也可以是单相交流 220 V，本例采用三相交流 220 V 供电，伺服驱动器的供电接线端子排是 CNP1。PLC 的高速输出点与伺服的 PP 端子连接，PLC 的输出和伺服驱动器的输入都是 NPN 型，因此是匹配的。PLC 的 1M 必须和伺服驱动器的 SG 连接，达到共地的目的。

【关键点】连线时，请注意 PLC 与伺服驱动器必须共地，否则不能形成回路；此外，三菱的伺服驱动器只能接受 NPN 信号，因此在选择 PLC 时，要注意选用 NPN 输出的 PLC，西门子的 S7－200 系列的 PLC 目前只有一款（CPU 224XPsi）是 NPN 输出。若读者一定要选用 PNP 输出的 PLC，则需要将信号进行转换，通常处理信号比较麻烦而且效果要差一些。

2. 伺服电动机的参数设定

用 PLC 的高速输出点控制伺服电动机，除了接线比用 PLC 的高速输出点控制步进电动机复杂外——后者不需要设置参数（细分的设置除外），还必须对伺服系统进行必要的参数设置。参数设置如下：

① P0＝0000，含义是位置控制，不进行再生制动。

② P3＝100，含义是齿轮比的分子。

③ P4＝1，含义是齿轮比的分母。

④ P41＝0，含义是伺服 ON、正行程限位和反行程限位都通过外部信号输入。

虽然伺服驱动器的参数很多，但对于简单的应用，只需要调整以上几个参数就足够了。

【关键点】设置完成以上参数后，不要忘记保存参数，伺服驱动器断电后，以上设置才起作用。此外，有的初学者编写程序时输入的脉冲数较少，而且齿轮比 P3/P4 又很小，发现系统运行后，伺服电动机并未转动，从而不知所措，其实伺服电动机已经旋转，只不过肉眼没有发现其转动，读者只要把输入的脉冲数增加到足够大，将齿轮比调大一些，就能发现伺服电动机旋转。

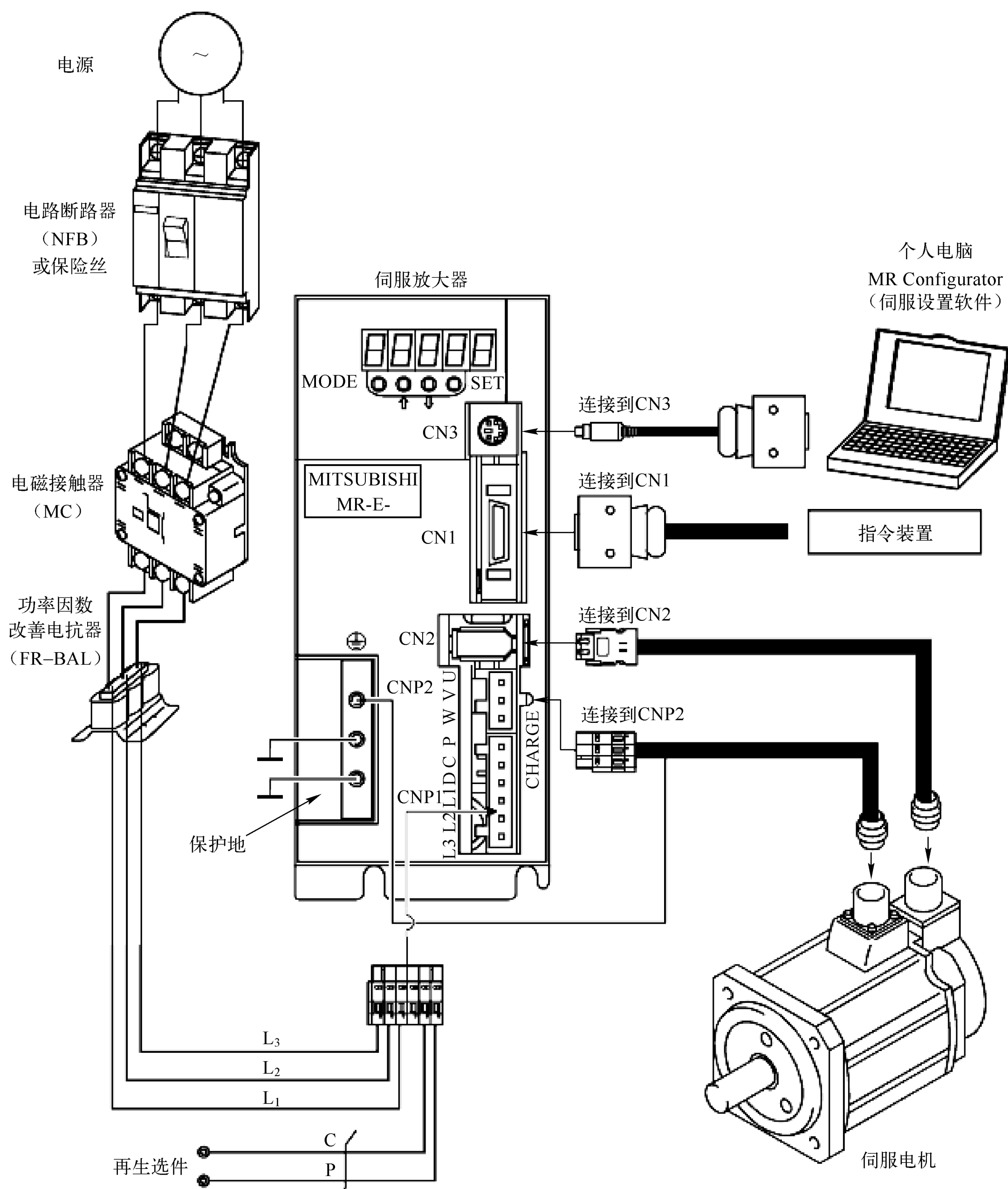


图 8–46　三菱 MR – E – A 的硬件连接

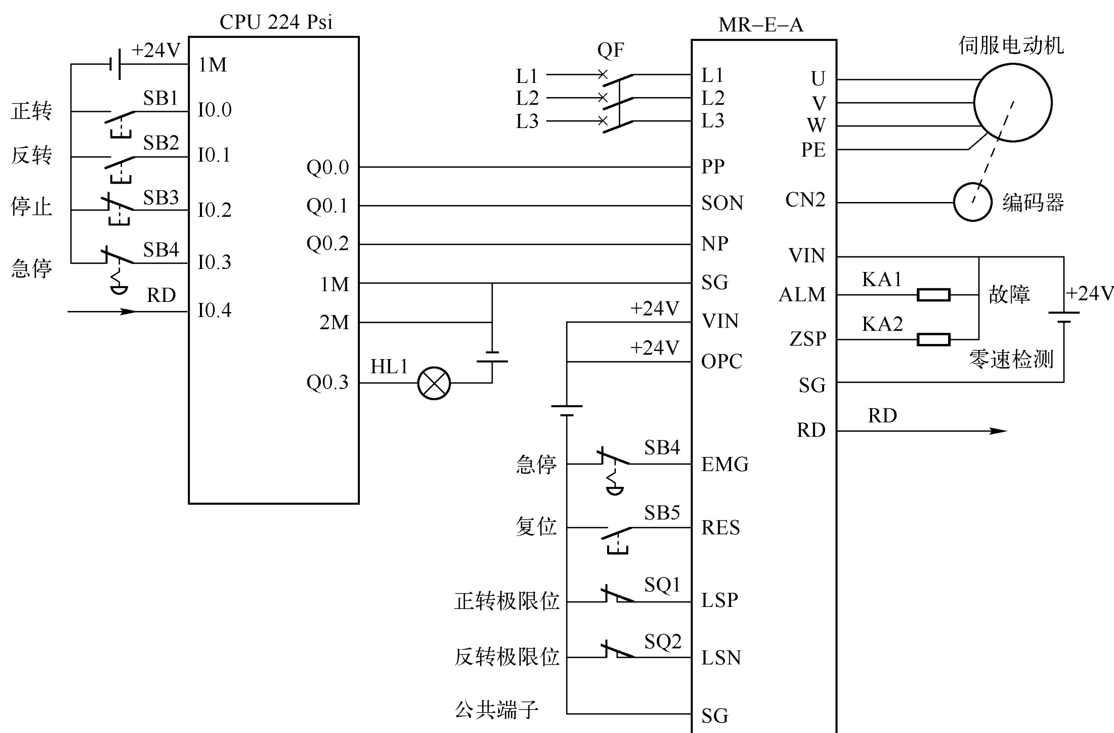


图 8-47　接线图

在前面的章节中介绍了直接使用 PLC 的高速输出点控制步进电动机，其实直接使用 PLC 的高速输出点控制伺服电动机的方法与之类似，只不过后者略微复杂一些，下面将用一个例子介绍具体的方法。

【例 8-8】 某设备上有一套伺服驱动系统，伺服驱动器的型号为 MR－E－A，伺服电动机的型号为 HF－KE13W1－S100，是三相交流同步伺服电动机，要求：压下按钮 SB1 时，伺服电动机带动系统 X 方向移动，，压下按钮 SB2 时，伺服电动机带动系统 X 负方向移动，当压下 SB3 伺服系统停机，伺服系统正常运行时，灯闪亮。请画出 I/O 接线图并编写程序。

【解】

1）主要软硬件配置：

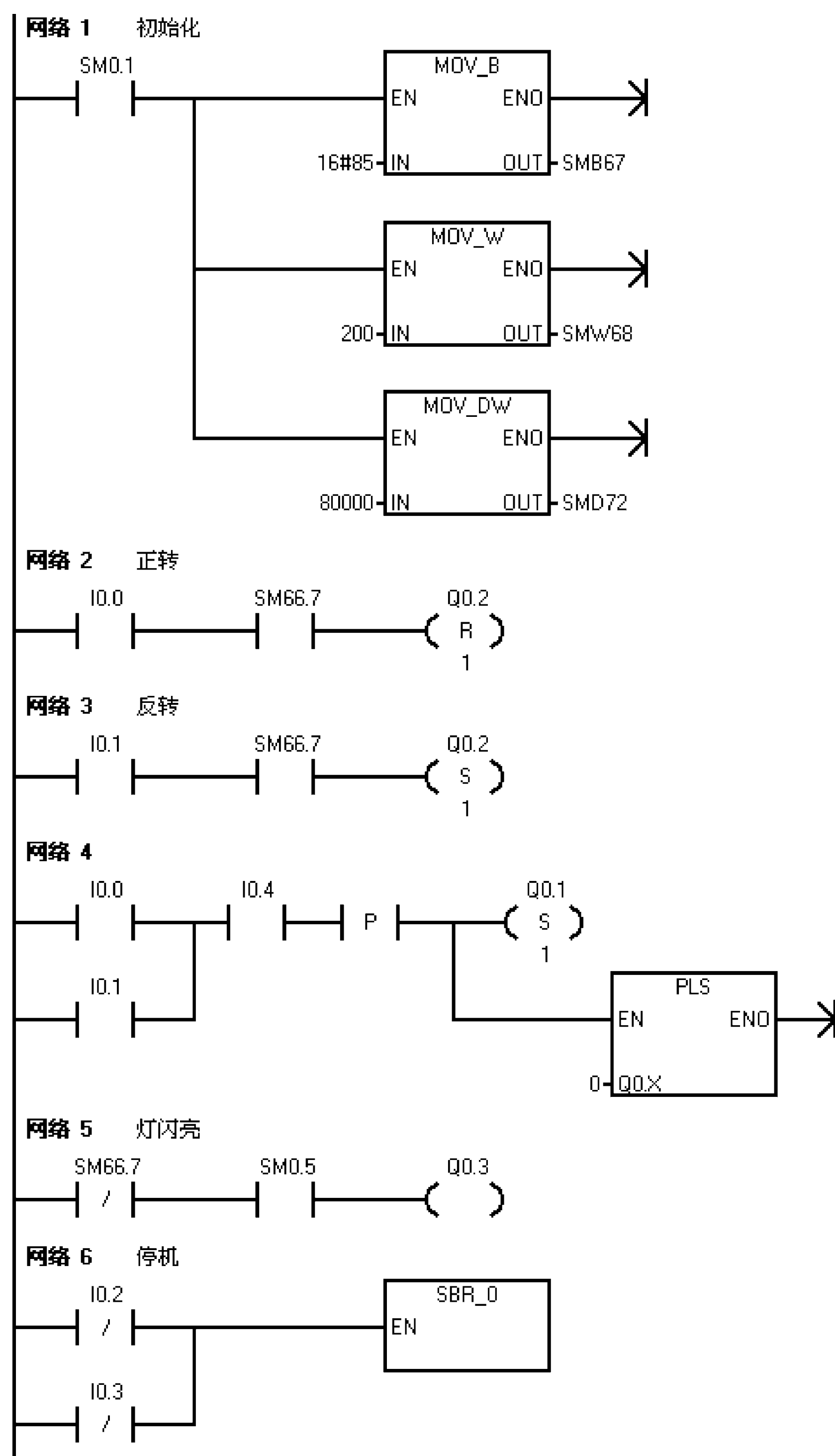
① 1 套 STEP 7－Micro/WIN V4.0 SP9。

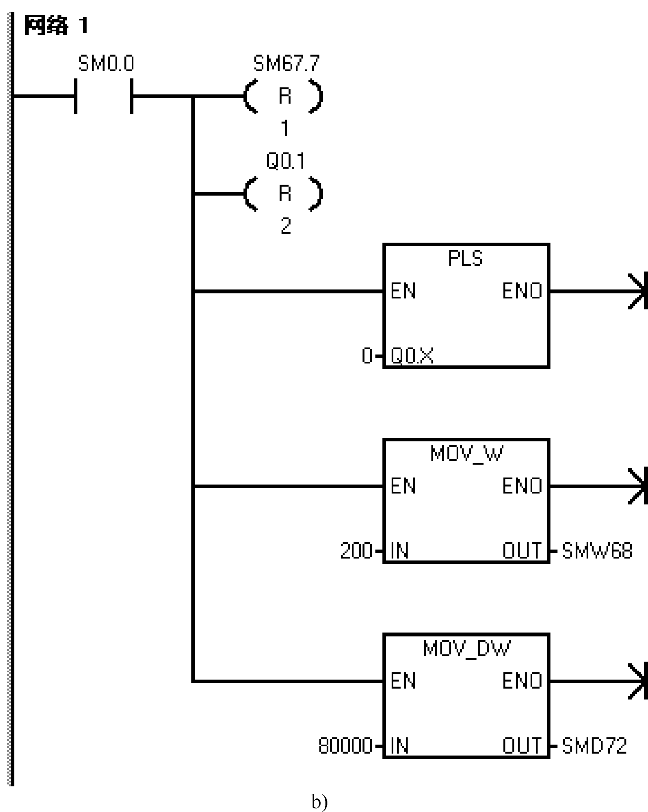
② 1 台伺服电动机，型号为 HF－KE13W1－S100。

③ 1 台伺服驱动器的型号为 MR－E－A。

④ 1 台 CPU 224XPsi 。

2）控制程序的编写。用 PLC 的高速输出点控制伺服电动机的程序与用 PLC 的高速输出点控制步进电动机的程序类似，这里不作过多的解释，其程序如图 8-48 所示。当完成系统接线、参数设定和程序下载后，当按下按钮 SB1 时，伺服电动机正转，当按下 SB3 伺服电动机停转，当按下 SB2 按钮伺服电动机反转。


网络 1 初始化
SM0.1
MOV_B
EN
ENO
16#85
IN
OUT
SMB67
MOV_W
EN
ENO
200
IN
OUT
SMW68
MOV_DW
EN
ENO
80000
IN
OUT
SMD72
网络 2 正转
I0.0
SM66.7
Q0.2
R
1
网络 3 反转
I0.1
SM66.7
Q0.2
S
1
网络 4
I0.0
I0.4
P
Q0.1
S
1
I0.1
PLS
EN
ENO
0
Q0.X
网络 5 灯闪亮
SM66.7
/
SM0.5
Q0.3
网络 6 停机
I0.2
/
SBR_0
EN
I0.3
/
a)



b)

图 8-48　PLC 的高速输出点控制伺服电动机程序

a）主程序　b）子程序

小　　结

（1）理解高速输出点的 PTO 控制寄存器的含义，这是编程的基础。

（2）位置向导的使用很方便，但对于其生成的子程序的含义要特别清楚，否则很难编写程序。

（3）无论是步进驱动系统还是伺服驱动系统的接线，都比以前学习的逻辑控制的接线要复杂很多，所以使用前一定要确保接线正确。

习　　题

1. 将步进电动机的红线和绿线对换会产生什么现象？

2. 步进电动机不通电时，用手可以拨动转轴（因为不带制动），那么通电后，不加信号时，用手能否拨动转轴？解释这个现象。

3. 有一台步进电动机，其脉冲当量是 3 度/脉冲，问此步进电动机转速为 250 r/min 时，转 10 圈，若用 CPU 226CN 控制，请画出接线图，并编写梯形图程序。

4. CPU 226CN 的输入端能否使用 +5 V 的电源？CPU 226CN 的输出端能否使用 +5 V 的

电源？CPU 224XP 的输入端能否使用 +5 V 的电源？

5. 用一台 CPU 222 和一只电感式接近开关测量一台电动机的转速，先设计接线图，再编写梯形图程序。

6. 实现一个简单的位置控制。控制要求：

① 用多齿凸轮与电动机联动，并用接近开关来检测多齿凸轮，产生的脉冲输入至 PLC 的计数器。

② 电动机转动至 4900 个脉冲时，使电动机减速，到 5000 个脉冲时，使电动机停止，同时剪切机动作将材料切断，并使脉冲计数复位。

第 9 章　PLC 在变频器调速系统中的应用

本章介绍电动机的特性和调速原理。西门子 MM 440 变频器的基本使用方法、PLC 控制变频器多段调速、PLC 控制变频器模拟量调速和 USS 通信调速。使用变频器时，三相异步电动机的制动和正反转控制。

9.1　三相异步电动机的机械特性和调速原理

9.1.1　三相异步电动机的机械特性

1. 三相异步电动机的机械特性

在异步电动机中，转速 $n=(1-S)\times n_0$，为了符合习惯画法，可将曲线换成转速与转矩之间的关系曲线，即称为异步电动机的机械特性，理解异步电动机的机械特性，是至关重要的，后续章节都会用到。公式如下：

$$T=\frac{km_1pU_1^2R_2S}{2\pi f_1[R_2^2+(SX_{20})^2]}=Km_1\Phi I_2\cos\varphi_2 \tag{9-1}$$

以上公式简化如下：

$$T=K\frac{SR_2U_1^2}{R_2^2+(SX_{20})^2}=K\frac{SR_2U^2}{R_2^2+(SX_{20})^2} \tag{9-2}$$

式中　K——与电动机结构参数、电源频率有关的一个常数；

U_1、U——定子绕组电压，电源电压；

R_2——转子每相绕组的电阻；

X_{20}——电动机不动（$S=1$）时转子每相绕组的感抗。

三相异步电动机的固有机械特性曲线如图 9-1所示。

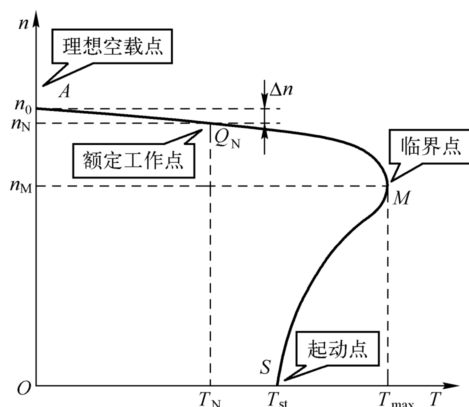


图 9-1　三相异步电动机的固有机械特性曲线

从特性曲线上可以看出，其上有四个特殊点可以决定特性曲线的基本形状和异步电动机的运行性能，这四个特殊点是：

1）$T=0, n=n_0, S=0$。电动机处于理想空载工作点，此时电动机的转速为理想空载转速。此时电动机的转速可以达到同步转速，即图中的 A 点，坐标为$(0, n_0)$。

2）$T=T_N, n=n_N, S=S_N$。电动机额定工作点，即图中的 Q_N点，坐标为（0，Q_N）。此时额

定转矩和额定转差率为：

$$T_N = 9.55\frac{P_N}{n_N}, S_N = \frac{n_0 - n_N}{n_0} \tag{9-3}$$

式中 P_N——电动机的额定功率；

n_N——电动机的额定转速，一般 $n_N = (0.94 \sim 0.985)n_0$；

S_N——电动机的额定转差率，一般 $S_N = 0.06 \sim 0.015$；

T_N——电动机的额定转矩。

3）$T = T_{st}, n = 0, S = 1$。电动机的起动工作点。电动机刚接通电源，但转速为0时，称为起动工作点，这时的转矩 T_{st} 称为起动转矩，也称堵转转矩，即图中的 S 点，坐标为（T_{st}，0）。起动转矩满足如下公式：

$$T_{st} = K\frac{R_2U^2}{R_2^2 + X_{20}^2} \tag{9-4}$$

可见：

① 异步电动机的起动转矩 T_{st} 与 U、R_2 及 X_{20} 有关。

② 当施加在定子每相绕组上的电压降低时，起动转矩会明显减小。

③ 当转子电阻适当增大时，起动转矩会增大。

④ 若增大转子电抗则会使起动转矩大为减小。

⑤ 一般情况下：$T_{st} \geqslant 1.5T_N$，这个数据是比较重要的。

4）$T = T_{max}, n = n_m, S = S_m$ 电动机的临界工作点。在这一点电动机产生的转矩最大，称为临界转矩 T_{max}，即图中的 M 点，坐标为（T_{max}, n_M）。临界转矩公式如下：

$$T_{max} = K\frac{U^2}{2X_{20}} \tag{9-5}$$

临界转矩与额定转矩之比就是异步电动机的过载能力，它表征了电动机能够承受冲击负载的能力大小，是电动机的又一个重要运行参数，一般过载能力 λ_m 大于等于2，即：

$$T_{max} = \lambda_m \cdot T_N \geqslant 2 \cdot T_N \tag{9-6}$$

9.1.2 三相异步电动机的调速原理

分析式（9-5）可知：异步电动机的机械特性与电动机的参数有关，也与外加电源电压 U、电源频率 f 有关，将关系式中的参数人为地加以改变而获得的特性称为异步电动机的人为机械特性。改变定子电压 U、定子电源频率 f、定子电路串入电阻或电抗、转子电路串入电阻或电抗，改变磁极对数等，都可得到异步电动机的人为机械特性。这就是异步电动机调速的原理。以下分别介绍。

1. 改变定子绕组电压调速

这种调速方式实际就是改变转差率调速。降低电动机电源电压 U 的人为特性，如当定子绕组外加电压为 U_N、$0.8U_N$、$0.5U_N$ 时，转子输出最大转矩分别为 $T_a = T_{max}$、$T_b = 0.64T_{max}$ 和 $T_c = 0.25T_{max}$。可见，电压愈低，人为特性曲线愈往左移，如图9-2所示。

由于 $T_{max} \propto U^2$ 和 $T_{st} \propto U^2$，所以当异步电动机的定子绕组电压降低时，起动转矩和临界转矩都会大幅度降低，而且机械特性会明显变软（所谓机械特性变软，就是指负载转矩增加时，电动机的转速降低增加；如果电动机机械特性变硬，指负载转矩增加时，电动机的转

速不降低或降低很少）。运行时，如电压降低太多，会大大降低它的过载能力与起动转矩，甚至使电动机发生带不动负载或者根本不能起动的现象。

此外，电网电压下降，在负载不变的条件下，将使电动机转速下降，转差率 S 增大，电流增加，引起电动机发热甚至烧坏。可见，降压调速，会降低起动转矩和临界转矩，并会使电动机的机械特性变软，其调速范围小，所以它并不是一种理想的调速方法。

【例 9-1】 电动机运行在额定负载 T_N 下，使 $\lambda_m=2$，若电网电压下降到 70% U_N，求 T_{max}。

【解】

$$T_{max}=\lambda_m T_N\left(\frac{U}{U_N}\right)^2=2\times0.7^2\times T_N=0.98T_N$$

2. 定子电路接入电阻 R_2 或电抗 X_2 时的人为特性

在电动机定子电路中外串电阻或电抗后，电动机端电压为电源电压减去定子外串电阻上或电抗上的压降，致使定子绕组相电压降低，这种情况下的人为特性与降低电源电压时的相似，在此不再赘述。其机械特性曲线如图 9-3 所示。

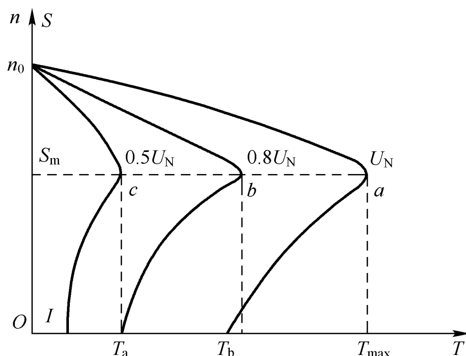


图 9-2　三相异步电动机的电压调速时机械特性曲线

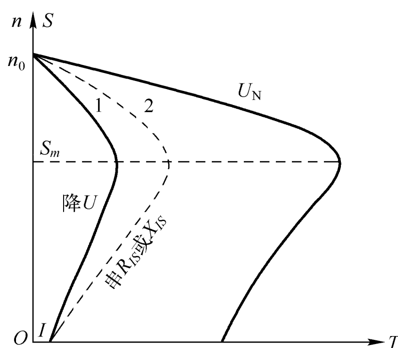


图 9-3　三相异步电动机的定子串电阻调速时机械特性曲线

3. 转子电路串电阻调速

转子电路串电阻调速，也称为变转差率调速。在三相线绕式异步电动机的转子电路中串入电阻后如图 9-4a 所示，转子电路中的电阻为 R_2+R_{2r}。

串电阻调速的特点：如图 9-4b 所示，串电阻后，临界转矩不变，但起动转矩增加；机械特性变软；低速时，调速范围小；是一种有级调速；转子电路串电阻调速的机械性能比定子串电阻要好，但这种调速方式仅用于绕线式电动机的调速，如起重机的电动机；低速时，能耗高。

4. 改变磁极对数调速

生产中，大量的生产机械并不需要连续平滑调速，只需要几种特定的转速，如只要求几种转速的有级变速的小功率机械，且对起动性能要求不高，一般只在空载或轻载起动选用变级变速电动机（双速、三速、四速）。

特点：体积大，结构简单；有级调速，调速范围小，最大传动比是 4；用于中小机床，替代齿轮箱，如早期的镗床。这种调速方式的使用在减少。

5. 定子电源的变频调速

（1）恒转矩调速

一般变频调速采用恒转矩调速，即希望最大转矩保持为恒值，为此在改变频率的同时，电源电压也要作相应的变化，使 U/f 为一个恒定值，这在实质上是使电动机气隙磁通保持不变。如图 9-5 所示，变频器在频率 f_1 和 f_2 工作时，就是恒转矩调速，这种调速方式中，保持 U/f 不变，临界转矩不变，起动转矩变大，机械硬度不变。又由于 $P=9.55\cdot T_{N}\cdot n$，电动机的输出功率随着其转速的升高，成比例升高。

（2）恒功率调速

当工作频率大于额定频率（如 $f_3>f$）时，变频器是恒功率调速。保持定子绕组的电压 U 不变，但磁通 φ_{m} 要减小，所以也叫弱磁调速。由公式 $T=9.55\frac{P_{N}}{n}$可知，采用恒功率调速时，随着转速的升高，电动机的输出转矩会降低，但机械硬度不变。可见，变频调速是一种理想的调速方式。毫无疑问，这种调速方式将越来越多被采用，是当前交流调速的主流。

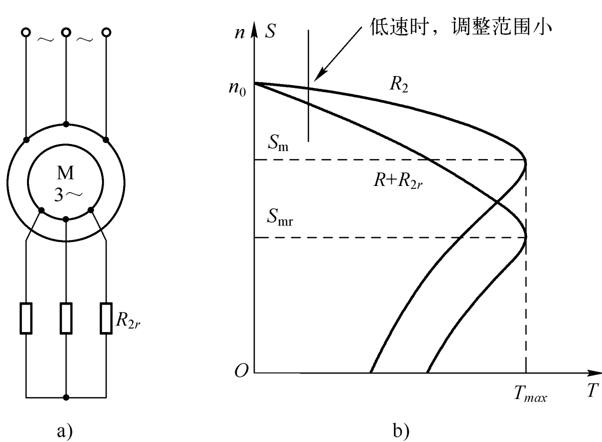


图 9-4　三相异步电动机的串电阻调速时机械特性曲线

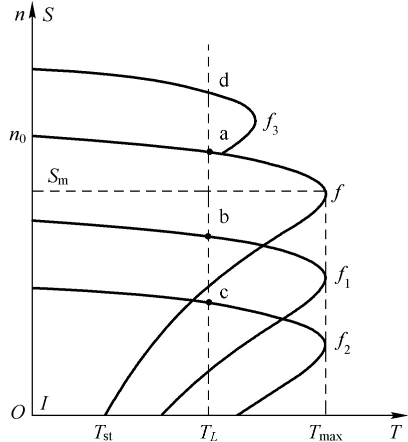


图9-5　三相异步电动机的变频调速时机械特性曲线

交流调速方式比较见表 9-1。

表 9-1　交流调速方式比较

调速方式名称	控 制 对 象	特　　点
变极调速	交流异步电动机	有级调速，系统简单，最多 4 段速
调压调速		无级调速，调速范围窄 电动机最大出力能力下降，效率低 系统简单，性能较差
转子串电阻调速		
变频调速	交流异步电动机 交流同步电动机	真正无级调速，调速范围宽 电动机最大出力能力不变，效率高 系统复杂，性能好 可以和直流调速系统相媲美

【例 9-2】某三相异步电动机，$P_M=60\,kW$，$n_N=750\,r/min$，$f_1=50\,Hz$，$\lambda_m=2.5$，试绘出电动机的固有机械特性。

【解】

1）同步转速点 A：$s=0$ 时，$n_N=750\ \mathrm{r/min}$，$T=0$

2）额定运行 Q_N 点：

$$s_N=\frac{n_1-n_N}{n_1}=\frac{750-725}{750}=0.033$$

$$T_N=9550\frac{P_N}{n_N}=9550\times\frac{60}{725}\mathrm{N\cdot m}=790\ \mathrm{N\cdot m}$$

3）最大转矩点 M：

$$s_m=s_N(\lambda_m+\sqrt{\lambda_m^2-1})=0.033(2.5+\sqrt{2.5^2-1})$$
$$=0.160$$

$$T_m=\lambda_m T_N=2.5\times790\ \mathrm{N\cdot m}=1975\ \mathrm{N\cdot m}$$

4）起动点 S。将 $s=1$ 时代入公式得：

$$T=\frac{2T_m}{\frac{s}{s_m}+\frac{s_m}{s}}=\frac{2\times1975}{\frac{1}{0.16}+\frac{0.16}{1}}\mathrm{N\cdot m}=616\ \mathrm{N\cdot m}$$

电动机的固有机械特性如图 9-6 所示。

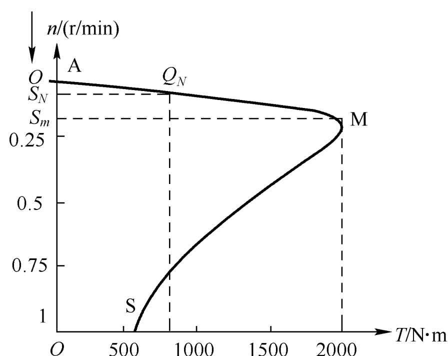


图 9-6　电动机的固有机械特性

9.1.3　变频器的发展

1. 变频器之前的几件大事

在讲解变频器的之前，有几件与之相关的大事不得不提，具体如下：

1）1831 年，英国的物理学家法拉第发现电磁感应原理，这使得人类使用电力成为可能。

2）1832 年，法国的皮克西研制出世界上第一台直流发电动机，这标志着电气时代的开始。

3）1873 年，比利时的古拉姆研制出世界上第一台直流电动机，结束了蒸汽机时代。

直流电动机调速方便，控制灵活。但直流电动机由于本身结构上存在有机械换向器和电刷，所以给直流调速系统带来了以下主要缺点：

① 维修困难。

② 使用环境受限制，不适用于易燃、易爆及环境恶劣的地方。

③ 制造大容量、高转速及高电压的直流电动机比较困难。

4）1882 年，塞尔维亚（后加入美国籍）的特斯拉继爱迪生发明直流电（DC）后不久，即发明了交流电（AC），并制造出世界上第一台交流电动机，并于 1888 年获得美国专利。但交流电动机的调速性能较差，促使研究交流系统的调速技术。

5）1920 年后即发现了变频调速的优越性。

6）20 世纪 60 年代电力电子技术得到快速发展，1957 年美国通用电气公司发明晶闸管，并于 1958 年投入商用，晶闸管的诞生为变频调速提供可能。

2. 变频器技术的发展阶段

芬兰瓦萨控制系统有限公司前身为瑞典的 STRONGB，于 20 世纪 60 年代成立，并于 1967 年开发出世界上第一台变频器，被称为变频器的鼻祖，开创了世界商用变频器的市场。之后变频器技术不断发展，如按照变频器的控制方式，可划分为以下几个阶段：

（1）第一阶段：恒压频比 U/F 技术

U/F 控制就是保证输出电压跟频率成正比的控制，这样可以使电动机的磁通保持一定，避免弱磁和磁饱和现象的产生，多用于风机、泵类节能型变频器。日本于 20 世纪 80 年代，开发出电压空间矢量控制技术，后引入频率补偿控制。电压空间矢量的频率补偿方法，不仅能消除速度控制的误差，而且可以通过反馈估算磁链幅值，消除低速时定子电阻的影响，将输出电压、电流闭环，以提高动态的精度和稳定度。

（2）第二阶段：矢量控制

20 世纪 70 年代，德国人（F. Blaschke）首先提出的矢量控制模型。矢量控制实现的基本原理是通过测量和控制异步电动机定子电流矢量，根据磁场定向原理分别对异步电动机的励磁电流和转矩电流进行控制，从而达到控制异步电动机转矩的目的。1992 年，西门子开发出 6SE 70 系列矢量控制的变频器，是矢量控制模型的代表产品。

矢量控制方式又有基于转差频率控制的矢量控制方式、无速度传感器矢量控制方式和有速度传感器的矢量控制方式等。这样就可以将一台三相异步电动机等效为直流电动机来控制，因而获得与直流调速系统同样的静、动态性能。矢量控制算法已被广泛地应用在 Siemens、ABB、GE、Fuji 和 SAJ 等国际化大公司变频器上。

（3）第三阶段：直接转矩控制

直接转矩控制系统简称 DTC（Direct Torque Control）是在 20 世纪 80 年代中期继矢量控制技术之后发展起来的一种高性能异步电动机变频调速系统。1977 年美国学者 A. B. Plunkett 在 IEEE 杂志上首先提出了直接转矩控制理论，1985 年由德国鲁尔大学 Depenbrock 教授和日本 Tankahashi 分别取得了直接转矩控制在应用上的成功，接着在 1987 年又把直接转矩控制推广到弱磁调速范围。不同于矢量控制，直接转矩控制具有鲁棒性强、转矩动态响应速度快、控制结构简单等优点，它在很大程度上解决了矢量控制中结构复杂、计算量大、对参数变化敏感等问题。直接转矩控制技术的主要问题是低速时转矩脉动大，其低速性能还是不能达到矢量控制的水平。

1995 年美国 ABB 公司推出 ACS 600 直接转矩控制系列变频器是直接转矩控制的代表产品。

20 世纪 60 年代到本世纪初，变频器技术发展的历程见表 9-2。

表 9-2　变频器技术发展的历程

项目	60 年代	70 年代	80 年代	90 年代	00 年代
电动机控制算法	U/F 控制		矢量控制	无速度矢量控制电流矢量 U/F	算法优化
功率半导体技术	SCR	GTR	IGBT	IGBT 大容量	更大容量更高开关频率
计算机技术			单片机 DSP	高速 DSP 专用芯片	更高速率和容量
PWM 技术		PWM 技术	SPWM 技术	空间电压矢量调制技术	PWM 优化新一代开关技术
变频器的特点	大功率传动使用变频器，体积大，价格高		变频器体积缩小，开始在中小功率电动机上使用	超静音变频器开始流行解决了 GTR 噪声问题，变频器性能大幅提升大批量使用，取代直流	未来发展方向完美无谐波，如：矩阵式变频器

3. 我国变频器技术发展现状

20 世纪 80 年代，大量中小型日本品牌的变频器进入我国市场。90 年代后，欧美大容量变频器如西门子、ABB、AB、科比、罗克韦尔等进入中国市场。

1989 年大连引入日本东芝第一条变频器生产线。

1991 年中国台湾普传进入内地，建了许多合资公司、研发中心和合资厂，是当时最大最完善的国内合资营销网，各合资公司组装变频器机箱，主控制板都是中国台湾做好后运来，pi89 变频器成了当时国内变频器市场上主打产品，但现已出局。

目前，国内有超过 100 多个厂家生产变频器，以华为、森兰、英威腾和汇川为代表，技术水平较接近世界先进水平，但总市场份额只有 10% 左右。我国国产变频器的生产，主要是交流 380 V 的中小型变频器，且大部分产品为低压，而高压大功率则很少，能够研制、生产、并提供服务的高压变频器厂商更少，不过是少数几个具备科研能力或资金实力强的企业。并且在技术方面，更是普遍采用 *U/F* 控制方式，对中、高压电动机进行的变频调速改造。我国高压变频器的品种和性能，还处于发展的初级阶段，仍需大量从国外进口。这一现状主要表现在以下方面：

1）国外各大品牌的产品，加快了占领国内市场的步伐并将产品本地化。我国目前变频器市场较大，仍有巨大的发展潜力。

2）多数企业不具有足够资金进行科研和规模化生产，生产工艺相对落后，产品的技术含量低，品质有待提高，但总体上价格低廉。

3）国内高压变频器尚未形成一套完备的标准，产品差异性大，需要进一步完备、完善高压变频器的标准，同时，国产高压变频器的功率等级较低，一般不超过 3 500 kW。

4）高压变频器周边产业少，不够发达，约束了高压变频器的发展速度，很多变频器中的主要功率器件，无法自行生产，如：驱动电路，电解电容等。

5）自主研发能力在逐步提高，与发达国家的技术差距在缩小，自主创新的技术和产品也逐步得到应用。

6）已经研制出具有瞬时掉电再恢复、故障再恢复等性能的变频器，同时正在研发能够进行四象限运行的高压变频器。

4. 变频器技术存在的问题

变频器在使用中存在的主要问题，是干扰问题。电网是一个非常复杂的结构，电网谐波是对变频器产生干扰的主要干扰源。谐波源的产生，主要有各种整流设备、交/直流互换设备、电子电压调整设备、非线性负载以及照明设备等等。这些设备在起动和工作的时候，产生一些对电网的冲击波（即电磁干涉），使得电网中的电压、电流的波形发生一定的畸变。对电网中的其他设备产生的这种谐波的影响，需要进行简单的处理，即在接入变频器处加装电源滤波器，滤去干扰波，使变频器尽可能小地受到电网中的这些谐波的影响，从而稳定工作。其次，另一种共模干涉，则通过变频器的控制线，对控制信号产生一定的干扰，影响其正常工作。

5. 变频器的发展趋势

随着节约环保型社会发展模式的提出，人们开始更多地关注生活环境的品质。节能型、低噪声变频器，是今后一段时间发展的一个总趋势。我国变频器的生产商家虽然不少，但是缺少统一的、具体的规范标准，使得产品差异性较大。且大部分采用了 *U/F* 控制和电压矢

量控制，其精度较低，动态性能也不高，稳定性能较差，这些方面与国外同等产品相比有一定的差距。就变频器设备来说，其发展趋势主要表现在以下方面：

1）变频器将朝着高压大功率和低压小功率、小型化、轻型化的方向发展。

2）工业高压大功率变频器、民用低压中小功率变频器潜力巨大。

3）目前，IGBT、IGCT、SGCT 仍将扮演着主要的角色。SCR、GTO 将会退出变频器市场。

4）无速度传感器的矢量控制、磁通控制和直接转矩控制等技术的应用，将趋于成熟。

5）全面实现数字化和自动化，参数自设定技术，过程自优化技术，故障自诊断技术。

6）高性能单片机的应用，优化了变频器的性能，实现了变频器的高精度和多功能。

7）相关配套行业正朝着专业化、规模化发展，社会分工逐渐明显。

8）伴随着节约型社会的发展，变频器在民用领域的使用，会逐步得到推广和应用。

9.1.4　变频器的分类

变频器发展到今天，已经研制了多种适合不同用途的变频器，变频器的种类比较多，以下详细介绍变频器的分类。

1. 按变换的环节分类

1）交 – 直 – 交变频器。

则是先把工频交流通过整流器变成直流，然后再把直流变换成频率电压可调的交流，又称间接式变频器，是目前广泛应用的通用型变频器。

2）交 – 交变频器。

即将工频交流直接变换成频率电压可调的交流，又称直接式变频器。主要用于大功率（500 kW 以上）低速交流传动系统中，目前已经在轧机、鼓风机、破碎机、球磨机和卷扬机等设备中应用。这种变频器既可用于异步电动机，也可以用于同步电动机的调速控制。

这两种变频器的比较见表 9–3。

表 9–3　交 – 直 – 交变频器和交 – 交变频器的比较

交 – 直 – 交变频器	交 – 交变频器
· 结构简单 · 输出频率变化范围大 · 功率因数高 · 谐波易于消除 · 可使用各种新型大功率器件	· 过载能力强 · 效率高输出波形好 · 输出频率低 · 使用功率器件多 · 输入无功功率大 · 高次谐波对电网影响大

2. 按直流电源性质分类

1）电压型变频器。电压型变频器特点是中间直流环节的储能元件采用大电容，负载的无功功率将由它来缓冲，直流电压比较平稳，直流电源内阻较小，相当于电压源，故称电压型变频器，常选用于负载电压变化较大的场合。这种变压器应用广泛。

2）电流型变频器。电流型变频器特点是中间直流环节采用大电感作为储能环节，缓冲无功功率，即扼制电流的变化，使电压接近正弦波，由于该直流内阻较大，故称电流源型变频器（电流型）。电流型变频器的特点（优点）是能扼制负载电流频繁而急剧的变化。常选

用于负载电流变化较大的场合。

3. 按照用途分类

可以分为通用变频器、高性能专用变频器、高频变频器、单相变频器和三相变频器等。此外，变频器还可以按输出电压调节方式分类，按控制方式分类，按主开关元器件分类，按输入电压高低分类。

4. 按变频器调压方法

按变频器调压方法有两种，具体如下：

1）PAM 变频器。它是一种通过改变电压源 U_d或电流源 I_d的幅值进行输出控制的。这种变频器已很少使用了。

2）PWM 变频器。它是在变频器输出波形的每半个周期分割成许多脉冲，通过调节脉冲宽度和脉冲周期之间的“占空比”调节平均电压，其等值电压为正弦波，波形较平滑。

5. 按控制方式分

1）U/f 控制变频器（VVVF 控制）。U/f 控制就是保证输出电压跟频率成正比的控制。低端变频器都采用这种控制原理。

2）SF 控制变频器（转差频率控制）。转差频率控制就是通过控制转差频率来控制转矩和电流。是高精度的闭环控制，但通用性差，一般用于车辆控制。与 U/f 控制相比，其加减速特性和限制过电流的能力得到提高。另外，它有速度调节器，利用速度反馈构成闭环控制，速度的静态误差小。

3）VC 控制变频器（Vectory Control 矢量控制）。矢量控制实现的基本原理是通过测量和控制异步电动机定子电流矢量，根据磁场定向原理分别对异步电动机的励磁电流和转矩电流进行控制，从而达到控制异步电动机转矩的目的。一般用在高精度要求的场合。

4）直接转矩控制。简单地说就是将交流电动机等效为直流电动机进行控制。

6. 按国际区域分类

1）国产变频器品牌：安邦信、汇川、浙江三科、欧瑞传动、森兰、英威腾、蓝海华腾、迈凯诺、伟创、易泰帝等，国产的品牌已经超过 200 家。

2）欧美变频器品牌：西门子、科比、伦茨、施耐德、ABB、丹佛斯、ROCKWELL、VACON、AB、西威。

3）日本变频器品牌：富士、三菱、安川、三垦、日立、欧姆龙、松下电器、松下电工、东芝、明电舍。

4）韩国变频器品牌：LG 、现代、三星、收获。

7. 按电压等级分类

1）高压变频器：3 kV、6 kV、10 kV。

2）中压变频器：660 V、1140 V 。

3）低压变频器：220 V、380 V。

8. 按电压性质分类

1）交流变频器：AC－DC－AC（交－直－交）、AC－AC（交－交）。

2）直流变频器：DC－AC（直－交）。

9.2 西门子 MM 440 变频器使用简介

9.2.1 认识变频器

1. 初识变频器

变频器一般是利用电力半导体器件的通断作用将工频电源变换为另一频率的电能控制装置。变频器有着“现代工业维生素”之称，在节能方面的效果不容忽视。随着各界对变频器节能技术和应用等方面认识的逐渐加深，我国变频器市场变得异常活跃。

变频器产生的最初目的是速度控制，应用于印刷、电梯、纺织、机床和生产流水线等行业。而目前相当多的运用是以节能为目的。由于中国是能源消耗大国，而中国的能源储备又相对贫乏，因此国家大力提倡各种节能措施，其中着重推荐了变频器调速技术。在水泵、中央空调等领域，变频器可以取代传统的通过限流阀和回流旁路技术，充分发挥节能效果；在火电、冶金、矿山、建材行业，高压变频调速的交流电动机系统的经济价值正在得以体现。

变频器是一种高技术含量、高附加值、高效益回报的高科技产品，符合国家产业发展政策。在过去的 20 年里，我国变频器行业从起步阶段到目前正逐步趋于成熟，发展十分迅速。进入 21 世纪以来，我国中、低压变频器市场的增长速度超过了 20%，远远大于近几年的 GDP 增长水平。

从产品优势角度看，通过高质量地控制电动机转速，可提高制造工艺水准。变频器不但有助于提高制造工艺水平，尤其在精细加工领域，可以有效节约电能。是目前最理想、最有前途的电动机节能设备。

从变频器行业所处的宏观环境看，无论是国家中、长期规划、短期的重点工程、政策法规、国民经济整体运行趋势，还是人们节能环保意识的增强、技术的创新、发展高科技产业的要求，从国家相关部委到各相关行业，变频器都受到了广泛的关注，市场吸引力巨大。西门子变频器外形如图 9-7 所示。



图 9-7　变频器外形

2. 交 – 直 – 交变频调速的原理

以图 9-8 说明交 – 直 – 交变频调速的原理。交 – 直 – 交变频调速就是变频器先将工频交流电整流成直流电，逆变器在微控制器（如 DSP）的控制下，将直流电逆变成不同频率

的交流电。目前市面上的变频器多是利用这种原理工作的。

图 9-8 中 R0 起限流作用，当 R、S、T 端子上的电源接通时，R0 接入电路，以限制起动电流。延时一段时间后，晶闸管 VT 导通，将 R0 短路，避免造成附加损耗。Rt 为能耗制动电阻，当制动时，异步电动机进入发动状态，逆变器向电容 C 反向充电，当直流回路的电压，即电阻 R1、R2 上的电压，升高到一定的值时（图中实际上测量的是电阻 R2 的电压），通过泵升电路使开关器件 Vb 导通，这样电容 C 上的电能就消耗在制动电阻 Rt 上。通常为了散热，制动电阻 Rt 安装在变频器外侧。电容 C 除了参与制动外，在电动机运行时，主要起滤波作用。顺便指出起滤波作用是电容器的变频器称为电压型变频器；起滤波作用是电感器的变频器称为电流型变频器，比较多见的是电压型变频器。

微控制器经运算输出控制正弦信号后，经过 SPWM（正弦脉宽调制）发生器调制，再由驱动电路放大信号，放大后的信号驱动 6 个功率晶体管，产生三相交流电压 U、V、W 驱动电动机运转。

【例 9-3】 如图 9-8 所示，若将变频器的动力线的输入和输出接反是否可行？若不可行有什么后果？

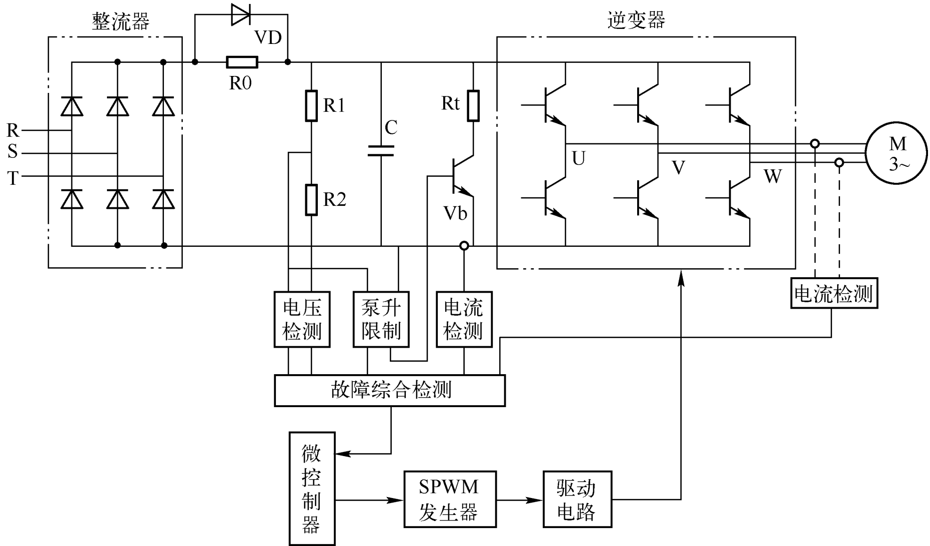


图 9-8　变频器原理图

【解】

将变频器的动力线的输入和输出接反是不允许的，可能发生爆炸。

9.2.2　西门子 MM 440 变频器使用简介

1. 初识西门子 MM 440 变频器

西门子 MM 440 变频器由微处理器控制，并采用具有现代先进技术水平的绝缘栅双极性晶体管（IGBT）作为功率输出器件，它具有很高的运行可靠性和功能多样性。脉冲宽度调制的开关频率也是可选的，降低了电动机的运行的噪声。

MM 440 变频器的框图如图 9-9 所示，控制端子定义见表 9-4。

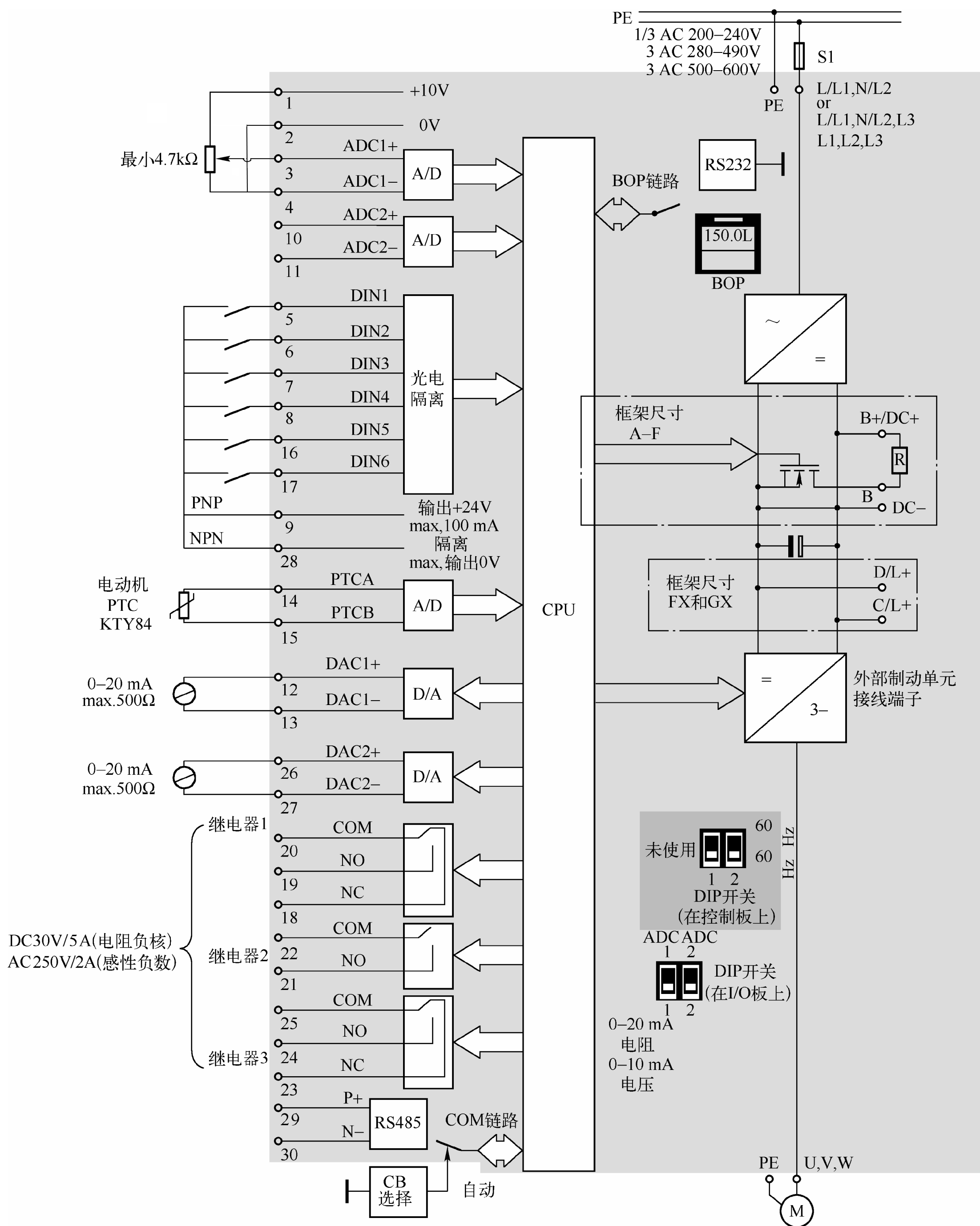


图 9-9　MM 440 变频器的框图

表 9-4 MM 440 控制端子表

端子序号	端子名称	功 能	端子序号	端子名称	功 能
1	-	输出 +10 V	16	DIN5	数字输入 5
2	-	输出 0 V	17	DIN6	数字输入 6
3	ADC1 +	模拟输入 1 （+）	18	DOUT1/NC	数字输出 1/常闭触点
4	ADC9 -	模拟输入 1 （-）	19	DOUT1/NO	数字输出 1/常开触点
5	DIN1	数字输入 1	20	DOUT1/COM	数字输出 1/转换触点
6	DIN2	数字输入 2	21	DOUT2/NO	数字输出 2/常开触点
7	DIN3	数字输入 3	22	DOUT2/COM	数字输出 2/转换触点
8	DIN4	数字输入 4	23	DOUT3NC	数字输出 3 常闭触点
9	-	隔离输出 +24 V / max. 100 mA	24	DOUT3NO	数字输出 3 常开触点
10	ADC2 +	模拟输入 2 （+）	25	DOUT3COM	数字输出 3 转换触点
11	ADC2 -	模拟输入 2 （-）	26	DAC2 +	模拟输出 2 （+）
12	DAC1 +	模拟输出 1 （+）	27	DAC2 -	模拟输出 2 （-）
13	DAC9 -	模拟输出 1 （-）	28	-	隔离输出 0 V/max. 100 mA
14	PTCA	连接 PTC/KTY84	29	P +	RS485
15	PTCB	连接 PTC/KTY84	30	P -	RS485

MM 440 变频器的核心部件是 CPU 单元，根据设定的参数，经过运算输出控制正弦波信号，再经过 SPWM 调制，放大输出正弦交流电驱动三相异步电动机运转。

MM 440 变频器是一个智能化的数字变频器，在基本操作板上可进行参数设置，参数可分为四个级别：

① 标准级，可以访问经常使用的参数。

② 扩展级，允许扩展访问参数范围，例如变频器的 I/O 功能。

③ 专家级，只供专家使用，即高级用户。

④ 维修级，只供授权的维修人员使用，具有密码保护。

【关键点】 对于一般的用户，将变频器设置成标准级或者扩展级即可。

BOP 基本操作面板的外形如图 9-10 所示，利用基本操作面板可以改变变频器的参数。BOP 具有 7 段显示的 5 位数字，可以显示参数的序号和数值，报警和故障信息，以及设定值和实际值。参数的信息不能用 BOP 存储。BOP 基本操作面板上的按钮的功能见表 9-5。

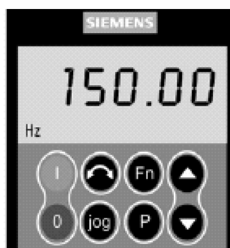


图 9-10 BOP 基本操作面板的外形

表 9-5　BOP 基本操作面板上的按钮的功能

显示/按钮	功　　能	功能的说明
P(1) r0000 Hz	状态显示	LED 显示变频器当前的设定值
I	起动变频器	按此键起动变频器。默认值运行时此键是被封锁的。为了使此键起作用，应设定 P0700 = 1
O	停止变频器	OFF1：按此键，变频器将按选定的斜坡下降速率减速停车；默认值运行时此键被封锁；为了允许此键起作用，应设定 P0700 = 1。OFF2：按此键两次（或一次，但时间较长）电动机将在惯性作用下自由停车，此功能总是“使能”
	改变电动机的旋转方向	按此键可以改变电动机的旋转方向。电动机的反向用负号（-）表示或用闪烁的小数点表示。在默认设定时此键被封锁。为使此键有效，应先按“起动电动机”键
jog	电动机点动	在“准备合闸”状态下按压此键，则电动机起动并运行在预先设定的点动频率。当释放此键，电动机停车。当电动机正在旋转时，此键无功能
Fn	功能	此键用于浏览辅助信息。 变频器运行过程中，在显示任何一个参数时按下此键并保持不动 2 s，将显示以下参数值（在变频器运行中，从任何一个参数开始）： ① 直流回路电压（用 d 表示，单位：V）。 ② 输出电流（A）。 ③ 输出频率（Hz）。 ④ 输出电压（用 o 表示，单位：V）。 ⑤ 由 P0005 选定的数值（如果 P0005 选择显示上述参数中的任何一个（3、4 或 5），这里将不再显示）。 连续多次按下此键，将轮流显示以上参数。 跳转功能。在显示任何一个参数（rXXXX 或 PXXXX）时，短时间按下此键，将立即跳转到 r0000，如果需要的话，可以接着修改其他的参数。跳转到 r0000 后，按此键将返回原来的显示点
P	访问参数	按此键即可访问参数
▲	增加数值	按此键即可增加面板上显示的参数数值
▼	减少数值	按此键即可减少面板上显示的参数数值
Fn + P	AOP 菜单	调出 AOP 菜单提示（仅用于 AOP）

2. MM 440 变频器 BOP 速度给定

以下用一个例子介绍 MM 440 变频器 BOP 速度给定的过程。

【例 9-4】 一台 MM 440 变频器配一台西门子三相异步电动机，已知电动机的技术参数，功率为 0.75 kW，额定转速为 1 380 r/min，额定电压为 380 V，额定电流为 2.05 A，额定频率为 50 Hz，试用 BOP 设定电动机的运行频率设定为 10 Hz。

【解】

1）先介绍如何设定参数，以下通过将参数 P1000 的第 0 组参数，即设置 P1000[0] = 1 的设置过程为例，讲解一个参数的设置方法。参数的设定方法见表 9-6。

表 9-6　参数的设定方法

序　号	操 作 步 骤	BOP 显示
1	按P键，访问参数	r0000
2	按▲键，直到显示 P1000	P1000
3	按P键，显示 in000，即 P1000 的第 0 组值	in000
4	按P键，显示当前值 2	2
5	按▼键，达到所要求的数值 1	1
6	按P键，存储当前设置	P1000
7	按Fn键，显示 r0000	r0000
8	按P键，显示频率	10.00

2）完整的设置过程。设置步骤见表 9-7。

表 9-7　设置步骤

步骤	参数及设定值	说　　明	步骤	参数及设定值	说　　明
1	P0003 = 2	扩展级	8	P1000 = 1	频率源为 BOP
2	P0010 = 1	为 1 才能修改电机参数	9	P0700 = 1	命令源（启停）为 BOP
3	P0304 = 380	额定电压	10	P1080 = 0	最小频率
4	P0305 = 2.05	额定电流	11	P1082 = 50	最大频率
5	P0307 = 0.75	额定功率	12	P1120 = 10	从静止到达最大频率所需时间
6	P0311 = 1 440	额定转速	13	P1121 = 10	从最大频率到停止所需时间
7	P0010 = 0	运行和设定变频器参数时，必须为 0			

3）起停控制。按下基本操作面板上的I按键，三相异步电动机起动，稳定运行的频率为 10 Hz；当按O按键时，电动机停机。

【关键点】 *初学者在设置参数时，有时不注意进行了错误的设置，但又不知道在什么参数的设置上出错，一般这种情况下可以对变频器进行复位，一般的变频器都有这个功能，复位后变频器的所有的参数变成出厂的设定值，但工程中正在使用的变频器要谨慎使用此功*

能。西门子 MM 440 的复位方法是，先将 P0010 设置为 30，再将 P0970 设置为 1，变频器上的显示器中闪烁的“busy”消失后，变频器成功复位。

【例 9-5】 如图 9-11 所示，问将输入端 L1 和 L2 电源线对调，三相交流电动机 M1 的转向是否会改变？为什么？若一定要通过“调线”的方法改变三相电动机的转向，又该怎么办？

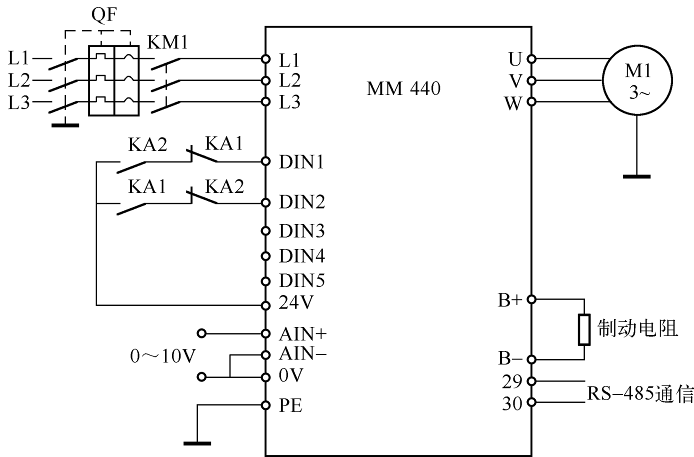


图 9-11　正反转原理图

【解】

1）将输入端 L1 和 L2 电源线对调，电动机的转向不会改变，因为这个操作不会改变变频器输出交流电的相序。

2）改变变频器输出端的 U、V、W 任意两根线都可改变电动机的转向。

顺便指出，通过变频器改变电动机的转向很容易，在后续章节将会介绍。

9.3　变频器多段速度给定

用基本操作面板进行手动速度给定方法简单，对资源消耗少，但这种速度给定方法对于操作者来说比较麻烦，而且不容易实现自动控制，而 PLC 控制的多段速度给定和通信速度给定，就容易实现自动控制，以下将用几个例题来介绍 MM 440 变频器的多段速度给定。

【例 9-6】 有一台 MM 440 变频器，接线图如图 9-12 所示，当按钮 SB1 时，三相异步电动机以 5 Hz 正转，当按钮 SB2 时，三相异步电动机以 10 Hz 正转，当按钮 SB3 时，三相异步电动机以 15 Hz 反转，已知电动机的技术参数，功率为 0.75 kW，额定转速为 1 440 r/min，额定电压为 380 V，额定电流为 2.05 A，额定频率为 50 Hz，请设计方案。

【解】

多段速度给定时，当按下按钮 SB1 时，DIN1 端子与变频器的 +24 V（端子 9）连接时对应一个频率，频率值设定在 P1001 中；当按下按钮 SB2 时，DIN2 端子与变频器的 +24 V（端子 9）连接时再对应一个频率，频率值设定在 P1002 中；当按下按钮 SB3 时，DIN3 端子与变频器的 +24 V 接通，对应一个频率，频率值设定在 P1003 中。变频器参数见表 9-8。

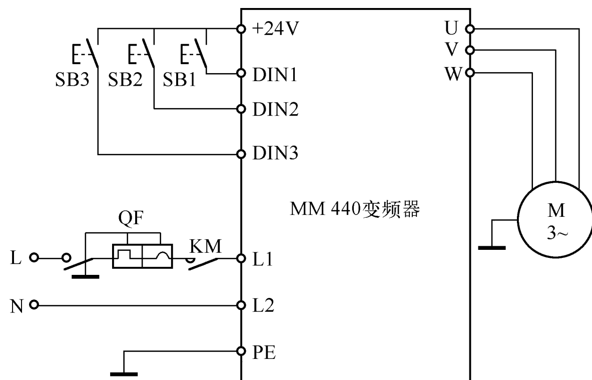


图 9-12　接线图

表 9-8　变频器参数

序号	变频器参数	出厂值	设定值	功 能 说 明
1	P0304	380	380	电动机的额定电压（380 V）
2	P0305	1.8	2.05	电动机的额定电流（2.05 A）
3	P0307	0.75	0.75	电动机的额定功率（0.75 W）
4	P0310	50	50	电动机的额定频率（50 Hz）
5	P0311	0	1 440	电动机的额定转速（1 440 r/min）
6	P1000	2	3	固定频率设定
7	P1080	0	0	电动机的最小频率（0 Hz）
8	P1082	50	50	电动机的最大频率（50 Hz）
9	P1120	10	10	斜坡上升时间（10 s）
10	P1121	10	10	斜坡下降时间（10 s）
11	P0700	2	2	选择命令源（由端子排输入）
12	P0701	1	16	固定频率设定值（直接选择 + ON）
13	P0702	12	16	固定频率设定值（直接选择 + ON）
14	P0703	9	2	反转
15	P1001	0	5	固定频率 1

【例 9-7】用一台继电器输出 CPU 226CN（DC/AC/Relay），控制一台 MM 440 变频器，当按钮 SB1 时，三相异步电动机以 5 Hz 正转，当按钮 SB2 时，三相异步电动机以 15 Hz 正转，当按钮 SB3 时，三相异步电动机以 15 Hz 反转，已知电动机的技术参数，功率为 0.06 kW，额定转速为 1 430 r/min，额定电压为 380 V，额定电流为 0.35 A，额定频率为 50 Hz，请设计方案，并编写程序。

【解】

1. 主要软硬件配置

① 1 套 STEP 7 - Micro/WIN V4.0 SP9。

② 1 台 MM 440 变频器。

③ 1 台 CPU 226CN。

④ 1 台电动机。

⑤ 1 根编程电缆（或者 CP 5611 卡）。

硬件接线图如图 9-13 所示。

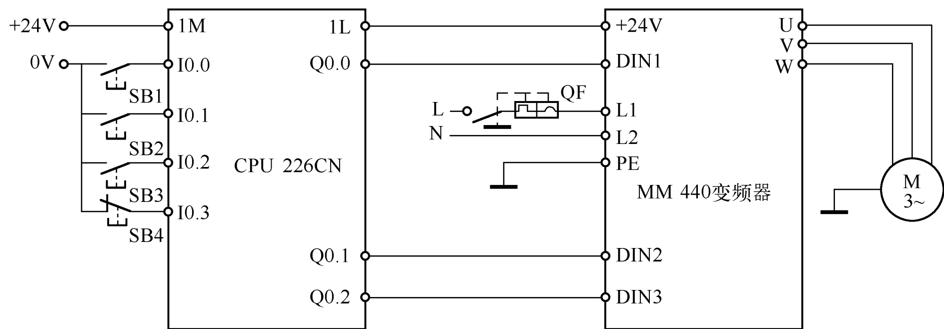


图 9-13　接线图（PLC 为继电器输出）

2. 参数的设置

多段速度给定时，当 DIN1 端子与变频器的 24 V（端子 9）连接时对应一个频率，当 DIN1 和 DIN2 端子同时与变频器的 24 V（端子 9）连接时再对应一个频率，DIN3 端子与变频器的 24 V 接通时为反转，DIN3 端子与变频器的 24 V 不接通时为正转。变频器参数见表 9-9。

表 9-9　变频器参数

序号	变频器参数	出厂值	设定值	功 能 说 明
1	P0304	230	380	电动机的额定电压（380 V）
2	P0305	1. 8	0. 35	电动机的额定电流（0. 35 A）
3	P0307	0. 75	0. 06	电动机的额定功率（60 W）
4	P0310	50	50	电动机的额定频率（50 Hz）
5	P0311	0	1430	电动机的额定转速（1 430 r/min）
6	P1000	2	3	固定频率设定
7	P1080	0	0	电动机的最小频率（0 Hz）
8	P1082	50	50	电动机的最大频率（50 Hz）
9	P1120	10	10	斜坡上升时间（10 s）
10	P1121	10	10	斜坡下降时间（10 s）
11	P0700	2	2	选择命令源（由端子排输入）
12	P0701	1	16	固定频率设定值（直接选择 + ON）
13	P0702	12	16	固定频率设定值（直接选择 + ON）
14	P0703	9	12	反转
15	P1001	0	5	固定频率 1
16	P1002	5	10	固定频率 2

当 Q0. 0 为 1 时，变频器的 9 号端子与 DIN1 端子连通，电动机以 5 Hz（固定频率 1）的转速运行，固定频率 1 设定在参数 P1001 中；当 Q0. 0 和 Q0. 1 同时为 1 时，DIN1 和 DIN2 端

子同时与变频器的24 V（端子9）连接，电动机以15 Hz（固定频率1 + 固定频率2）的转速运行，固定频率2设定在参数P1002中。

修改参数P0701，对应设定数字输入1（DIN1）的功能；修改参数P0702，对应设定数字输入2（DIN2）的功能，依次类推。

【关键点】 不管是什么类型PLC，只要是继电器输出，其接线图都可以参考图9-5，若增加三个中间继电器则更加可靠，如图9-14所示。

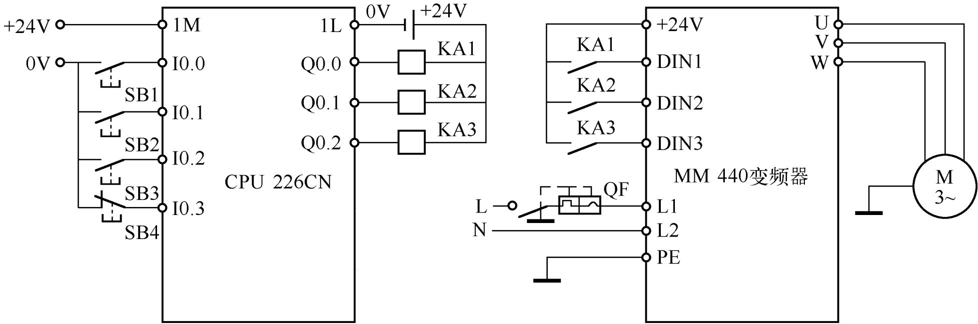


图9-14　接线图（PLC为继电器输出）

3. 编写程序

这个程序相对比较简单，如图9-15所示。

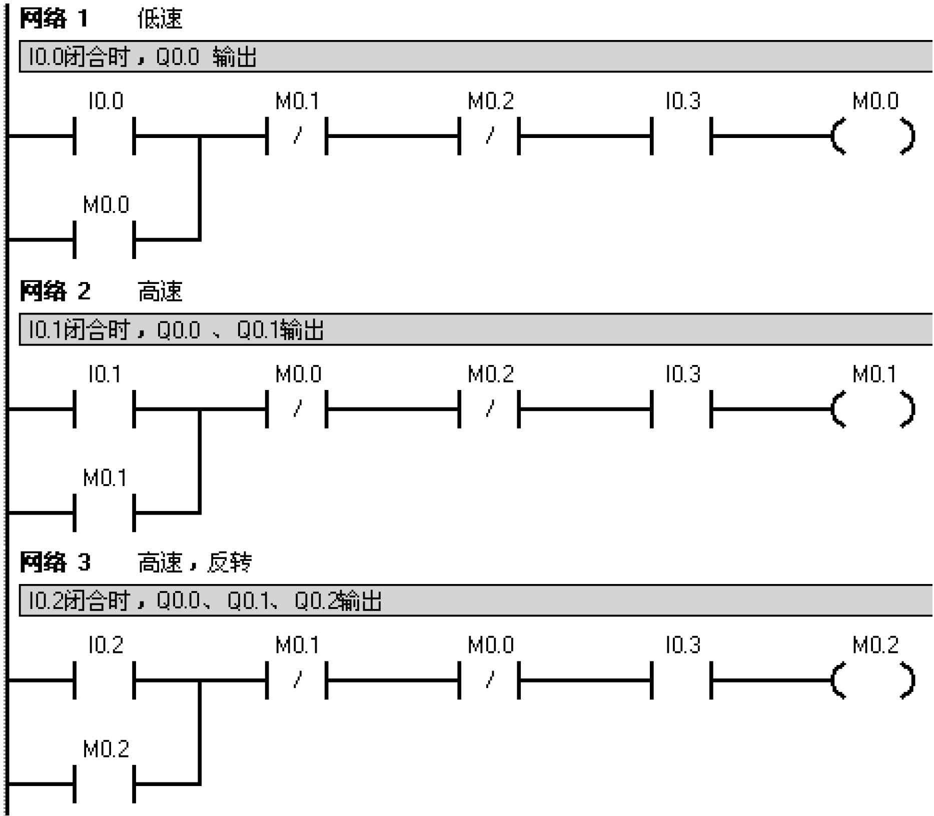


图9-15　程序

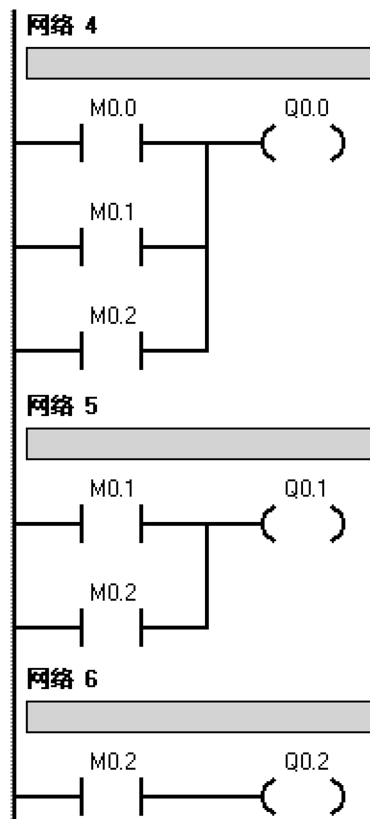


图 9-15　程序（续）

4. PLC 为晶体管输出（PNP 型输出）时的控制方案

西门子的 S7－200 系列 PLC 大多为 PNP 输出（目前只有 1 款为 NPN 输出），MM 440 变频器的默认为 PNP 输入，因此电平是可以兼容的。由于 Q0. 0（或者其他输出点输出时）输出的其实就是 DC 24 V 信号，又因为 PLC 与变频器有共同的 0 V，所以，当 Q0. 0（或者其他输出点输出时）输出时，就等同于 DIN1（或者其他数字输入）与变频器的 9 号端子（24 V）连通，硬件配置如图 9-16 所示，控制程序与图 9-15 中的相同。

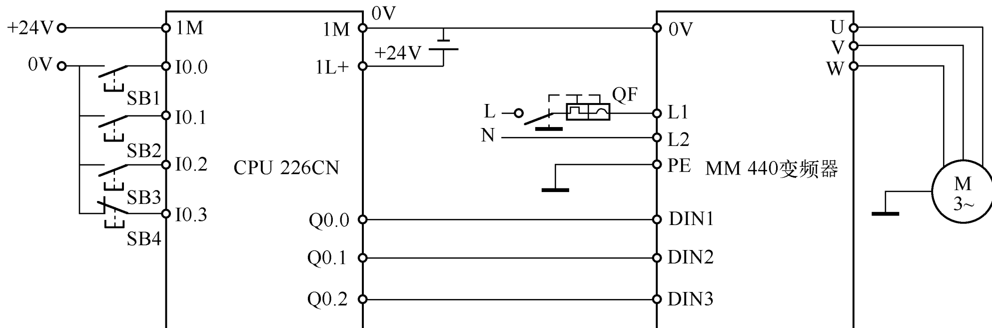


图 9-16　接线图（PLC 为 PNP 晶体管输出）

【关键点】PLC 为晶体管时，其 1M（0 V）必须与变频器的 0 V 短接，否则，PLC 的输出不能形成回路。

5. PLC 为晶体管输出（NPN 型输出）时的控制方案

日系的 PLC 晶体管输出多为 NPN 型，如三菱的 FX 系列 PLC（新型的 FX3U 也有 PNP 输出）多为 NPN 输出，而西门子 MM 440 变频器默认为 PNP 输入，显然电平是不匹配。所幸的是，西门子提供了解决方案，只要将参数 P0725 设置成 0（默认为 1），MM 440 变频器就变成 NPN 输入，这样就与 FX 系列 PLC 的电平匹配了。接线（PLC 为 NPN 晶体管输出）如图 9-17 所示。

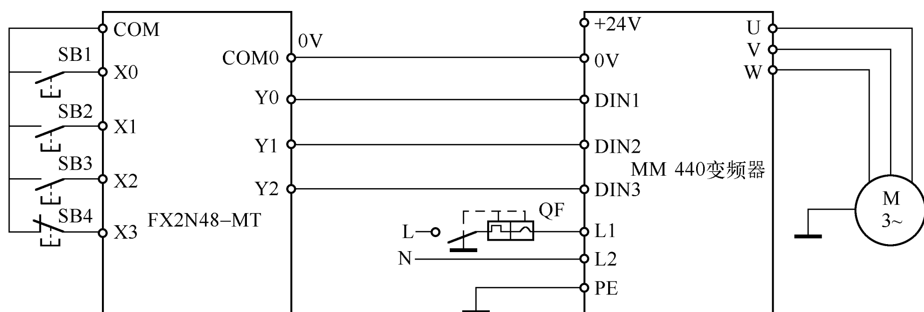


图 9-17　接线图（PLC 为 NPN 晶体管输出）

【关键点】 必须将参数 P0725 设置成 0（默认为 1），MM 440 变频器就才能变成 NPN 输入，这样 MM 440 变频器就与 FX 系列 PLC 的电平匹配了。有的变频器的输入电平的选择是通过跳线的方式实现的，如三菱的变频器。

6. S7-200（晶体管输出）控制三菱变频器的方案

西门子 S7-200 系列 PLC 大多为 PNP 输出（目前只有 1 款为 NPN 输出），三菱 A 740 变频器的默认为 NPN 输入，因此电平是不兼容的。而三菱变频器的输入电平也是输入和输出可以选择的，与西门子不同的是，需要将电平选择的跳线改换到 PNP 输入，而不需要改变参数设置。其接线图如图 9-18 所示。

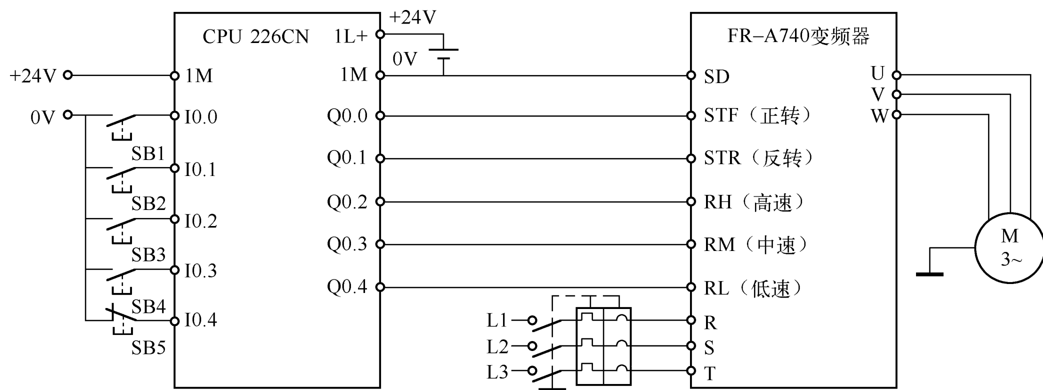
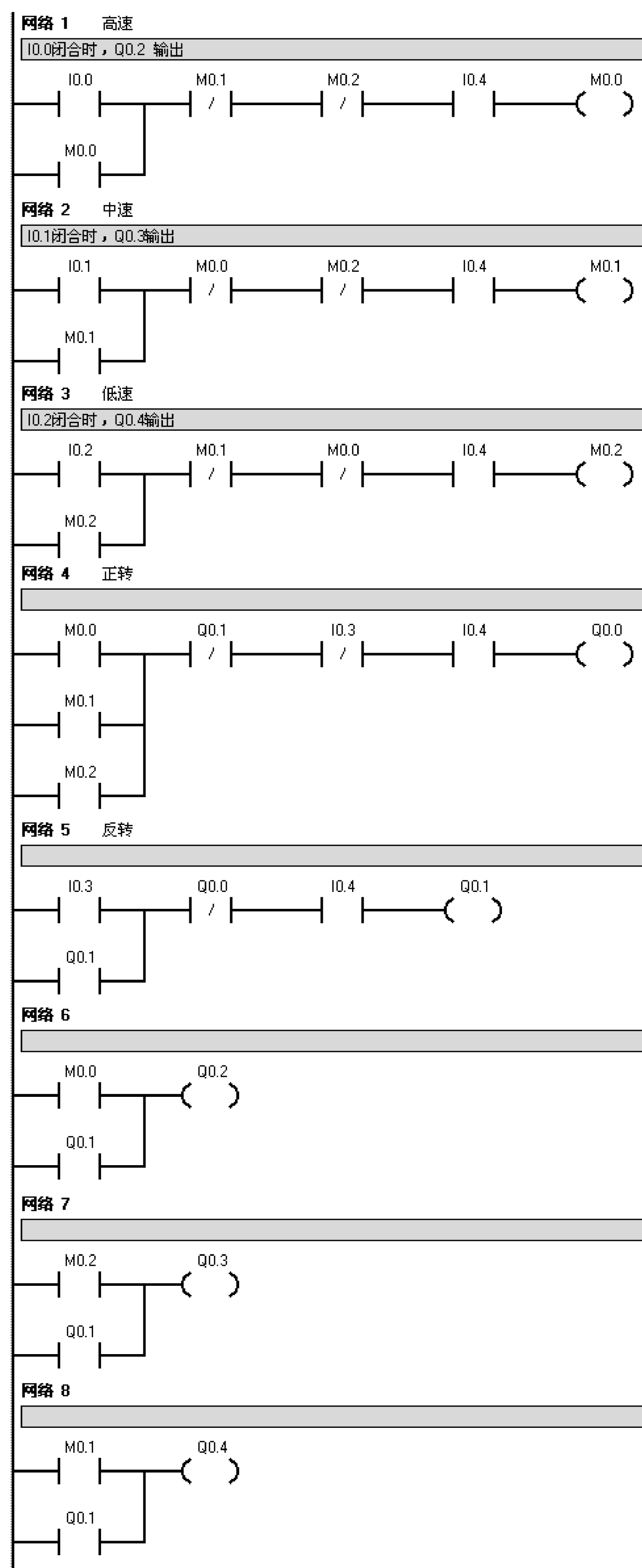


图 9-18　接线图（S7-200（晶体管输出）PLC，三菱 A740 变频器）

【关键点】 将电平选择的跳线改换到 PNP 输入（由默认的“SINK”改成“SOURCE”）。此外，接线图要正确。三菱的强电输入接线端子（R、S、T）和强电输出端子（U、V、W）相距很近，接线时，切不可接反。

当三菱 A740 变频器的 STF 高电平时，电动机正转；STR 高电平时，电动机反转；RH 高电平时，电动机高速运行（15 Hz），RL 高电平时，电动机低速运行（5 Hz），程序如图 9-19所示。


网络 1 高速
I0.0闭合时，Q0.2 输出
I0.0
M0.1
M0.2
I0.4
M0.0
M0.0
网络 2 中速
I0.1闭合时，Q0.3输出
I0.1
M0.0
M0.2
I0.4
M0.1
M0.1
网络 3 低速
I0.2闭合时，Q0.4输出
I0.2
M0.1
M0.0
I0.4
M0.2
M0.2
网络 4 正转
M0.0
Q0.1
I0.3
I0.4
Q0.0
M0.1
M0.2
网络 5 反转
I0.3
Q0.0
I0.4
Q0.1
Q0.1
网络 6
M0.0
Q0.2
Q0.1
网络 7
M0.2
Q0.3
Q0.1
网络 8
M0.1
Q0.4
Q0.1
图 9-19 程序

9.4　变频器模拟量速度给定

9.4.1　模拟量模块的简介

1. 模拟量 I/O 扩展模块的规格

模拟量 I/O 扩展模块包括模拟量输入模块、模拟量输出模块和模拟量输入输出模块。部分模拟量模块的规格见表 9-10。

表 9-10　模拟量 I/O 扩展模块规格表

模块型号	输入点	输出点	电压	功耗/W	电源要求	
					DC 5 V	DC 24 V
EM 232	0	2	DC 24 V	2	20 mA	70 mA
EM 235	4	1	DC 24 V	2	30 mA	60 mA

2. 模拟量 I/O 扩展模块的接线

S7 -200 系列的模拟量模块用于输入和输出电流或者电压信号。模拟量输出模块的接线如图 9-20 所示。

模拟量输入模块有两个参数容易混淆，即模拟量转换的分辨率和模拟量转换的精度（误差）。分辨率是 A - D 模拟量转换芯片的转换精度，即用多少位的数值来表示模拟量。若 S7 -200 模拟量模块的转换分辨率是 12 位，能够反映模拟量变化的最小单位是满量程的 1/4096。模拟量转换的精度除了取决于 A - D 转换的分辨率，还受到转换芯片的外围电路的影响。在实际应用中，输入的模拟量信号会有波动、噪声和干扰，内部模拟电路也会产生噪声、漂移，这些都会对转换的最后精度造成影响。这些因素造成的误差要大于 A - D 芯片的转换误差。

当模拟量的扩展模块的输入点/输出点有信号输入或者输出时，LED 指示灯不会亮，这点与数字量模块不同，因为西门子模拟量模块上的指示灯没有与电路相连。

使用模拟量模块时，要注意以下问题：

① 模拟量模块有专用的扁平电缆与 CPU 通信，并通过此电缆由 CPU 向模拟量模块提供 DC 5 V 的电源。此外，模拟量模块必须外接 DC 24 V 电源。

② 每个模块能同时输入/输出电流或者电压信号。双极性就是信号在变化的过程中要经过“0”，单极性不过“0”。由于模拟量转换为数字量是有符号整数，所以双极性信号对应的数值会有负数。在 S7 -200 中，单极性模拟量输入/输出信号的数值范围是 0 ~ 32 000；双极性模拟量信号的数值范围是 -32 000 ~ +32 000。

③ 一般电压信号比电流信号容易受干扰，应优先选用电流信号。电压型的模拟量信号，由于输入端的内阻很高（S7 -200 的模拟量模块为 10 MΩ），极易引入干扰。一般电压信号是用在控制设备柜内电位器设置，或者距离非常近、电磁环境好的场合。电流型信号不容易受到传输线沿途的电磁干扰，因而在工业现场获得广泛的应用。电流信号可以传输比电压信号远得多的距离。

④ 对于模拟量输出模块，电压型和电流型的输出信号的接线不同，各自的负载接到各

自的端子上。

⑤ 模拟量输出模块总是要占据两个通道的输出地址。即便有些模块（EM 235）只有一个实际输出通道，它也要占用两个通道的地址。在编程计算机和 CPU 实际联机时，使用 Micro/WIN 的菜单命令“PLC→信息（Information）”，可以查看 CPU 和扩展模块的实际 I/O 地址分配。

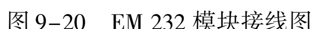


图 9-20　EM 232 模块接线图

9.4.2　模拟量速度给定的应用

数字量多段速度给定可以设定速度段是有限的，而且不能做到无级调速，而外部模拟量输入可以做到无级调速，也容易实现自动控制，而且模拟量可以是电压信号或者电流信号，使用比较灵活，因此应用较广。以下用 2 个例子介绍电流信号速度给定。

【例 9-8】要对一台变频器进行电压信号模拟量速度给定，已知电动机的技术参数，功率为 0.75 kW，额定转速为 1 440 r/min，额定电压为 380 V，额定电流为 2.05 A，额定频率为 50 Hz。设计电气控制系统，并设定参数。

【解】

电气控制系统如图 9-21 所示，只要调节电位器就可以实现对电动机进行无级调速，参数设定见表 9-11。

图 9-21　电气控制系统图

表 9-11　变频器参数表

序号	变频器参数	出厂值	设定值	功 能 说 明
1	P0304	380	380	电动机的额定电压（380 V）
2	P0305	1. 8	2. 05	电动机的额定电流（2. 05 A）
3	P0307	0. 75	0. 75	电动机的额定功率（0. 75 kW）
4	P0310	50	50	电动机的额定频率（50 Hz）
5	P0311	0	1 440	电动机的额定转速（1 440 r/min）
6	P0700	2	2	选择命令源（由端子排输入）
7	P0756	0	0	选择 ADC 的类型（电压信号）
8	P1000	2	2	频率源（模拟量）
9	P701	1	1	数字量输入 1

【例 9-9】用一台触摸屏、PLC 对变频器进行速度给定，已知电动机的技术参数，功率为 0. 06 kW，额定转速为 1430 r/min，额定电压为 380 V，额定电流为 0. 35 A，额定频率为 50 Hz。

【解】

1. 软硬件配置

① 1 套 STEP 7 - Micro/WIN V4. 0 SP9。

② 1 台 MM 440 变频器。

③ 1 台 CPU 226CN。

④ 1 台电动机。

⑤ 1 根编程电缆（或者 CP 5611 卡）。

⑥ 1 台 EM 232。

⑦ 1 台 HMI。

将 PLC、变频器、模拟量输出模块 EM 232 和电动机按照如图 9-22 所示接线。

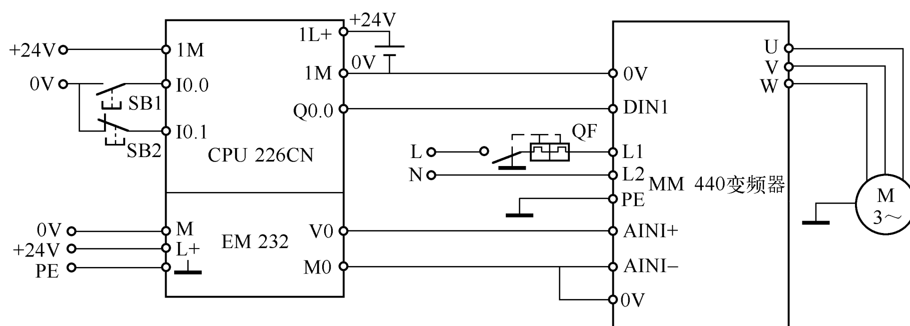


图 9-22　接线图

【关键点】接线时一定要把变频器的 0V 和 AIN1 - 短接，PLC 的 1M 与变频器的 0V 也要短接，否则不能进行调速。

2. 设定变频器的参数

查询 MM 440 变频器的说明书，变频器中参数设定见表 9 - 12。

表 9-12　变频器参数表

序号	变频器参数	出厂值	设定值	功能说明
1	P0304	230	380	电动机的额定电压（380 V）
2	P0305	3. 25	0. 35	电动机的额定电流（0. 35 A）
3	P0307	0. 75	0. 06	电动机的额定功率（60 W）
4	P0310	50	50	电动机的额定频率（50 Hz）
5	P0311	0	1 430	电动机的额定转速（1 430 r/min）
6	P0700	2	2	选择命令源（由端子排输入）
7	P0756	0	1	选择 ADC 的类型（电流信号）
8	P1000	2	2	频率源（模拟量）
9	P701	1	1	数字量输入 1

【**关键点**】P0756 设定成 1 表示电流信号对变频器进行速度给定，这是容易忽略的，默认是电压信号，这个参数的默认值为 0；此外，若调速模拟量是电流信号，还要将 I/0 控制板上的 DIP 开关设定为“ON”，如图 9-23 所示。本例的调速模拟量为电压信号，故不需要进行以上设置。

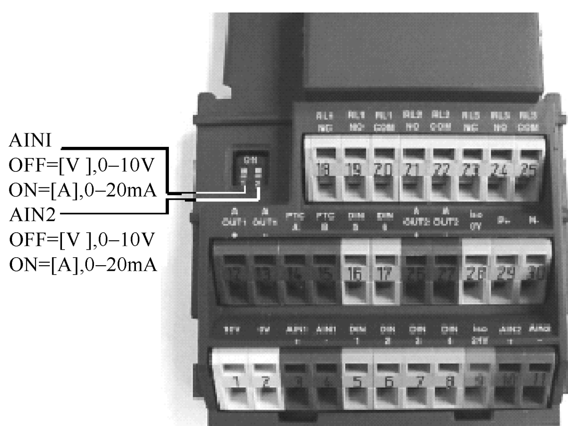


图 9-23　I/0 控制板上的 DIP 开关设定为“ON”

3. 编写程序，并将程序下载到 PLC 中

梯形图如图 9-24 所示。

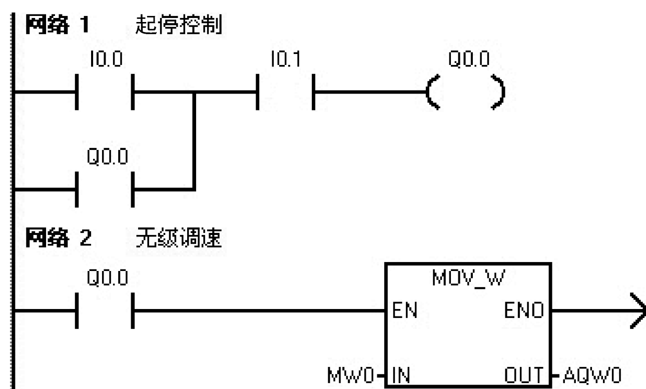


图 9-24　梯形图

9.5 变频器的通信速度给定

9.5.1 USS 协议简介

USS 协议（Universal Serial Interface Protocol——通用串行接口协议）是 Siemens 公司所有传动产品的通用通信协议，它是一种基于串行总线进行数据通信的协议。USS 协议是主－从结构的协议，规定了在 USS 总线上可以有一个主站和最多 31 个从站；总线上的每个从站都有一个站地址（在从站参数中设定），主站依靠它识别每个从站；每个从站也只对主站发来的报文做出响应并回送报文，从站之间不能直接进行数据通信。另外，还有一种广播通信方式，主站可以同时给所有从站发送报文，从站在接收到报文并做出相应的响应后，可不回送报文。

1. 使用 USS 协议的优点

1）对硬件设备要求低，减少了设备之间的布线。

2）无需重新连线就可以改变控制功能。

3）可通过串行接口设置以改变传动装置的参数。

4）可实时监控传动系统。

2. USS 通信硬件连接注意要点

1）条件许可的情况下，USS 主站尽量选用直流型的 CPU（针对 S7－200 系列）。

2）一般情况下，USS 通信电缆采用双绞线即可（如常用的以太网电缆）。如果干扰比较大，可采用屏蔽双绞线。

3）在采用屏蔽双绞线作为通信电缆时，把具有不同电位参考点的设备互连，造成在互连电缆中产生不应有的电流，从而造成通信口的损坏。所以要确保通信电缆连接的所有设备，共用一个公共电路参考点，或是相互隔离的，以防止不应有的电流产生。屏蔽线必须连接到机箱接地点或 9 针连接插头的插针 1。建议将传动装置上的 0 V 端子连接到机箱接地点。

4）尽量采用较高的波特率，通信速率只与通信距离有关，与干扰没有直接关系。

5）终端电阻的作用是用来防止信号反射的，并不用来抗干扰。如果在通信距离很近、波特率较低或点对点通信的情况下，可不用终端电阻。多点通信的情况下，一般也只需在 USS 主站上加终端电阻就可以取得较好的通信效果。

6）当使用交流型的 CPU 22X 和单相变频器进行 USS 通信时，CPU 22X 和变频器的电源必须接成同相位。

7）建议使用 CPU 226（或 CPU 224＋EM 277）来调试 USS 通信程序。

8）不要带电插拔 USS 通信电缆，尤其是正在通信过程中，这样极易损坏传动装置和 PLC 的通信端口。如果使用大功率传动装置，即使传动装置掉电后，也要等几分钟，让电容放电后，再去插拔通信电缆。

S7－200 利用 USS 通信速度给定，STEP 7－Micro/WIN V4.0 SP9 软件中必须另外安装指令库，因为指令库不是 STEP 7－Micro/WIN V4.0 SP9 的标准配置，需要购买。

9.5.2 USS 通信的应用

以下用一个例子介绍 USS 通信的应用。

【例 9-10】 用一台 CPU 226CN 对变频器进行 USS 无级速度给定，已知电动机的技术参数，功率为 0.06 kW，额定转速为 1440 r/min，额定电压为 380 V，额定电流为 0.35 A，额定频率为 50 Hz。请制定解决方案。

【解】

1. 软硬件配置

① 1 套 STEP 7 - Micro/WIN V4.0 SP9（含指令库）。

② 1 台 MM 440 变频器。

③ 1 台 CPU 226CN。

④ 1 台电动机。

⑤ 1 根编程电缆（或者 CP 5611 卡）。

⑥ 1 根屏蔽双绞线。

硬件配置如图 9-25 所示。

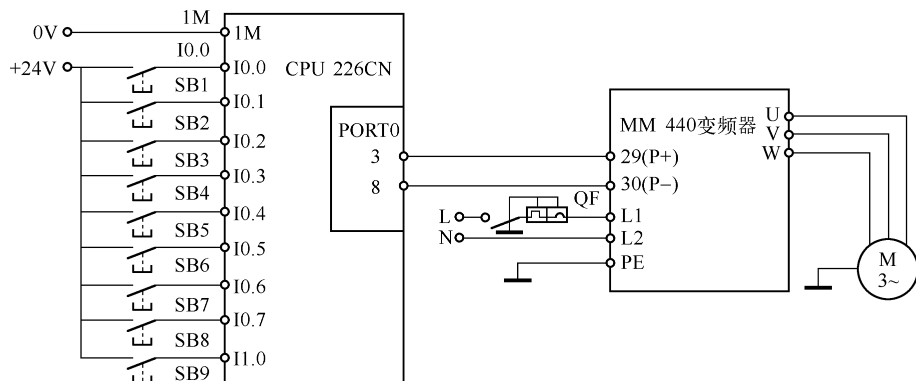


图 9-25　硬件配置图

【关键点】

1）图 9-25 中，编程口 PORT0 的第 3 脚与变频器的 29 脚相连，编程口 PORT0 的第 8 脚与变频器的 30 脚相连，并不需要占用 PLC 的输出点。还有一点要指出，STEP 7 - Micro/WIN V4.0 SP5 以前的版本中，USS 通信只能用 PORT0 口，而 STEP 7 - Micro/WIN V4.0 SP5（含）之后的版本，则 USS 通信可以用 PORT0 口和 PORT1 口。调用不同的通信口使用的子程序也不同。

2）图 9-25 中，串口的第 3 脚与变频器的 29 脚相连，串口的第 8 脚与变频器的 30 脚相连，并不需要占用 PLC 的输出点。图 9-25 的 USS 通信连接是要求不严格时的做法，一般的工业现场不宜采用，工业现场的 PLC 端应使用专用的网络连接器，且终端开组要接通，如图 9-26 所示，变频器端的连接图如图 9-27 所示，在购买变频器时附带有所需的电阻，并不需要另外购置。还有一点必须指出：如果有多台变频器，则只有最末端的变频器需要接入如图 9-27 的电阻。

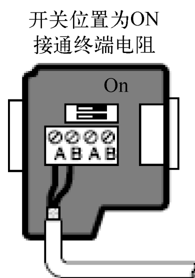


图 9-26　网络连接器图（PLC 端）

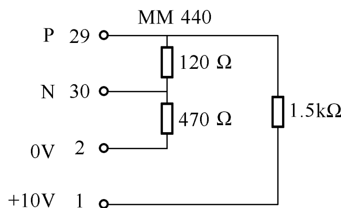


图 9-27　连接图（变频器端）

2. 相关指令介绍

（1）初始化指令

USS_INIT 指令被用于启用和初始化或禁止驱动器通信。在使用任何其他 USS 协议指令之前，必须执行 USS_INIT 指令，且无错。一旦该指令完成，立即设置“完成”位，才能继续执行下一条指令。

EN 输入打开时，在每次扫描时执行该指令。仅限为通信状态的每次改动执行一次 USS_INIT指令。使用边缘检测指令，以脉冲方式打开 EN 输入。欲改动初始化参数，执行一条新 USS_INIT 指令。USS 输入数值选择通信协议：输入值 1 将端口 0 分配给 USS 协议，并启用该协议；输入值 0 将端口 0 分配给 PPI，并禁止 USS 协议。BAUD（波特率）将波特率设为 1200、2400、4800、9600、19200、38400、57600 或 115200。

ACTIVE（激活）表示激活驱动器。当 USS_INIT 指令完成时，DONE（完成）输出打开。“错误”输出字节包含执行指令的结果。USS_INIT 指令格式见表 9-13。

表 9-13　USS_INIT 指令格式

LAD	输入/输出	含　　义	数 据 类 型
USS_INIT EN Mode　Done Baud　Error Active	EN	使能	BOOL
	Mode	模式	BYTE
	Baud	通信的波特率	DWORD
	Active	激活驱动器	DWORD
	Done	完成初始化	BOOL
	Error	错误代码	BYTE

站点号具体计算对应关系见表 9-14。

表 9-14　站点号具体计算对应关系

D31	D30	D29	D28	…	D19	D18	D17	D16	…	D3	D2	D1	D0
0	0	0	0		0	1	0	0		0	0	0	0

D0 ~ D31 代表 32 台变频器，要激活某一台变频器，就将该位置 1，上面的表格将 18 号变频器激活，其 16 进制表示为 16#00040000。若要将所有 32 台变频器都激活，则 ACTIVE 为 16#FFFFFFFF。

假如 PLC 只要和地址为 0 的变频器通信，那么 USS_INIT 指令中的“Active”怎样设置？

根据表 9-14，且 PLC 只要和地址为 0 的变频器通信，那么只要将 D0 = 1 即可，所以其 16

进制表示为 16#00000001。初学者很容易将之设置为 16#00000000，这是要特别注意的。

（2）控制指令

USS_CTRL 指令被用于控制 ACTIVE（激活）驱动器。USS_CTRL 指令将选择的命令放在通信缓冲区中，然后送至编址的驱动器（DRIVE（驱动器）参数），条件是已在 USS_INIT 指令的 ACTIVE（激活）参数中选择该驱动器。仅限为每台驱动器指定一条 USS_CTRL 指令。USS_CTRL 指令格式见表 9-15。

表 9-15　USS_CTRL 指令格式

LAD	输入/输出	含　义	数据类型
USS_CTRL EN RUN OFF2 OFF3 F_ACK　Resp_R DIR　Error Status Speed Drive　Run_EN Type　D_Dir Speed~　Inhibit Fault	EN	使能	BOOL
	RUN	模式	BOOL
	OFF2	允许驱动器滑行至停止	BOOL
	OFF3	命令驱动器迅速停止	BOOL
	F_ACK	故障确认	BOOL
	DIR	驱动器应当移动的方向	BOOL
	Drive	驱动器的地址	BYTE
	Type	选择驱动器的类型	BYTE
	Speed_SP	驱动器速度	DWORD
	Resp_R	收到应答	BOOL
	Error	通信请求结果的错误字节	BYTE
	Speed	全速百分比	DWORD
	Status	驱动器返回的状态字原始数值	WORD
	D_Dir	表示驱动器的旋转方向	BOOL
	inhibit	驱动器上的禁止位状态	BOOL
	Fault	故障位状态	BOOL

具体描述如下：

① EN 位必须打开，才能启用 USS_CTRL 指令。该指令应当始终启用。RUN（运行）（RUN/STOP（运行/停止））表示驱动器是打开（1）还是关闭（0）。当 RUN（运行）位打开时，驱动器收到一条命令，按指定的速度和方向开始运行。为了使驱动器运行，必须符合三个条件，分别是 DRIVE（驱动器）在 USS_INIT 中必须被选为 ACTIVE（激活）；OFF2 和 OFF3 必须被设为 0；FAULT（故障）和 INHIBIT（禁止）必须为 0。

② 当 RUN（运行）关闭时，会向驱动器发出一条命令，将速度降低，直至电动机停止。OFF2 位被用于允许驱动器滑行至停止。OFF3 位被用于命令驱动器迅速停止。Resp_R（收到应答）位确认从驱动器收到应答。对所有的激活驱动器进行轮询，查找最新驱动器状态信息。每次 S7 -200 从驱动器收到应答时，Resp_R 位均会打开，进行一次扫描，所有以下数值均被更新。F_ACK（故障确认）位被用于确认驱动器中的故障。当 F_ACK 从 0 转为 1 时，驱动器清除故障。DIR（方向）位表示驱动器应当移动的方向。“驱动器”（驱动器地址）输入是驱动器的地址，向该地址发送 USS_CTRL 命令。有效地址：0 ~31。“类型”（驱动器类型）输入选择驱动器的类型。将 3（或更早版本）驱动器的类型设为 0。将 4 驱动器的类型设为 1。

③ Speed_SP（速度设定值）是作为全速百分比的驱动器速度。Speed_SP 的负值会使驱动器反向旋转方向。范围：-200.0% ~200.0%。

④ Error 是一个包含对驱动器最新通信请求结果的错误字节。USS 指令执行错误标题定义可能因执行指令而导致的错误条件。

⑤ Status 是驱动器返回的状态字原始数值。

⑥ Speed 是作为全速百分比的驱动器速度。范围：-200.0% ~200.0%。

⑦ Run_EN（运行启用）表示驱动器是运行（1）还是停止（0）。

⑧ D_Dir 表示驱动器的旋转方向。

⑨ inhibit 表示驱动器上的禁止位状态（0 - 不禁止，1 - 禁止）。欲清除禁止位，“故障”位必须关闭，RUN（运行）、OFF2 和 OFF3 输入也必须关闭。

⑩ Fault 表示故障位状态（0 - 无故障，1 - 故障）。驱动器显示故障代码。欲清除故障位，纠正引起故障的原因，并打开 F_ACK 位。

3. 设置变频器的参数

先查询 MM 440 变频器的说明书，再依次在变频器中设定表 9-16 中的参数。

表 9-16　变频器参数表

序号	变频器参数	出厂值	设定值	功 能 说 明
1	P0304	230	380	电动机的额定电压（380 V）
2	P0305	3.25	0.35	电动机的额定电流（0.35 A）
3	P0307	0.75	0.06	电动机的额定功率（60 W）
4	P0310	50	50	电动机的额定频率（50 Hz）
5	P0311	0	1 440	电动机的额定转速（1 430 r/min）
6	P0700	2	5	选择命令源（COM 链路的 USS 设置）
7	P1000	2	5	频率源（COM 链路的 USS 设置）
8	P2010	6	6	USS 波特率（6 ~9600）
9	P2011	0	18	站点的地址
10	P2012	2	2	PZD 长度
11	P2013	127	127	PKW 长度（长度可变）
12	P2014	0	0	看门狗时间

【关键点】P2011 设定值为 18，与程序中的地址一致，正确设置变频器的参数是 USS 通信成功的前提。此外，要选用 USS 通信的指令，只要双击在如图 9-28 所示库中对应的指令即可。

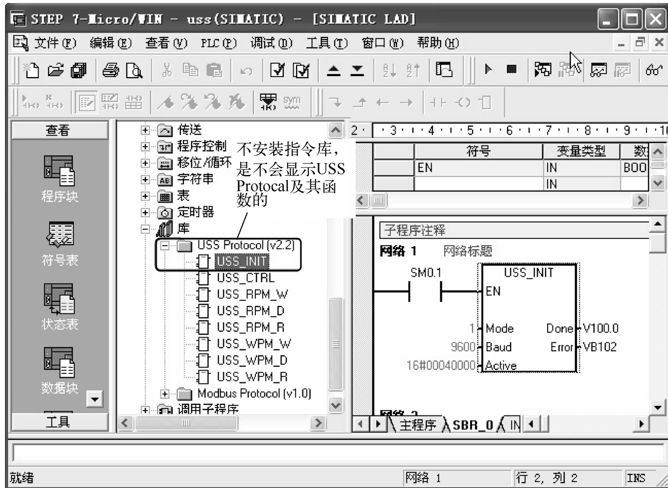


图 9-28　USS 指令库

4. 编写程序

程序如图 9–29 所示。

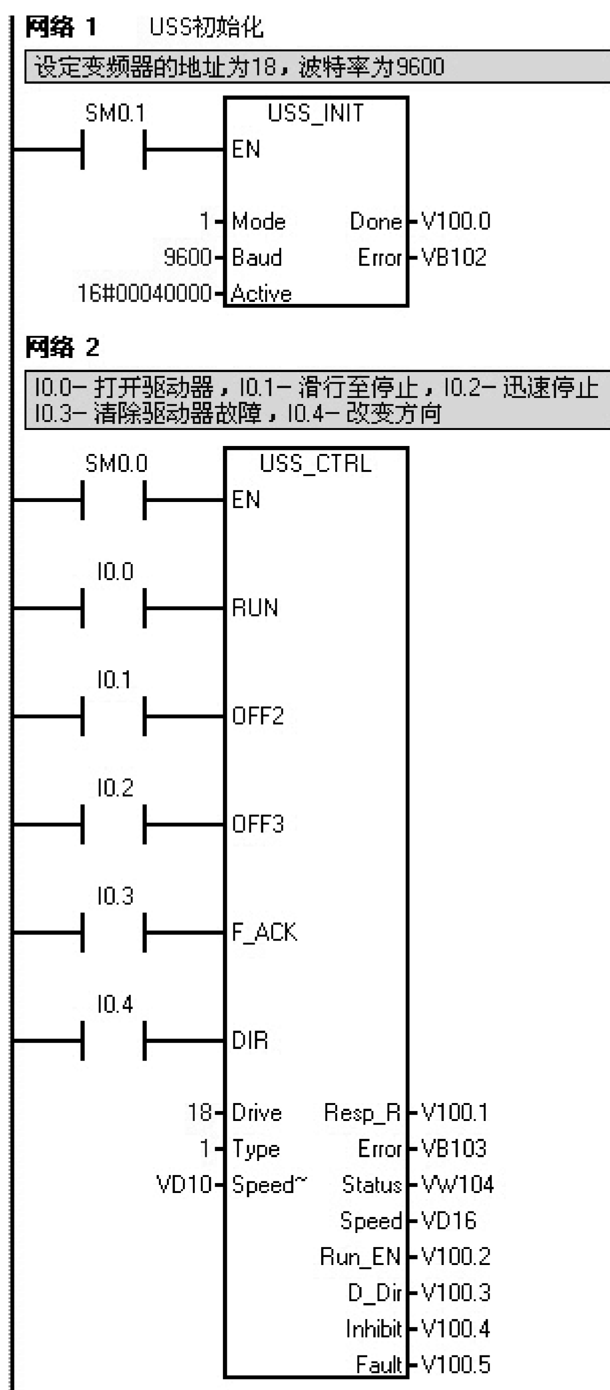


图 9–29　程序

9.6　使用变频器时的常用回路

9.6.1　制动控制回路

1. 制动的含义

提供反向转矩，使电动机停止或减速。制动主要有直流制动、电阻制动、回馈制动和电磁抱闸制动。以下只介绍电阻制动。

2. 电阻制动

在某些应用场合，电动机会工作于发电状态，例如：起重机、牵引电动机、将货物向下运送的传送带。在发电状态下，电能将通过逆变器回馈到直流环节，将导致直流环节 PN 之间的电压上升。当电压上升到一定程度时功率管 VT 导通，回馈的能量消耗在电阻上，原理如图 9-30 所示。

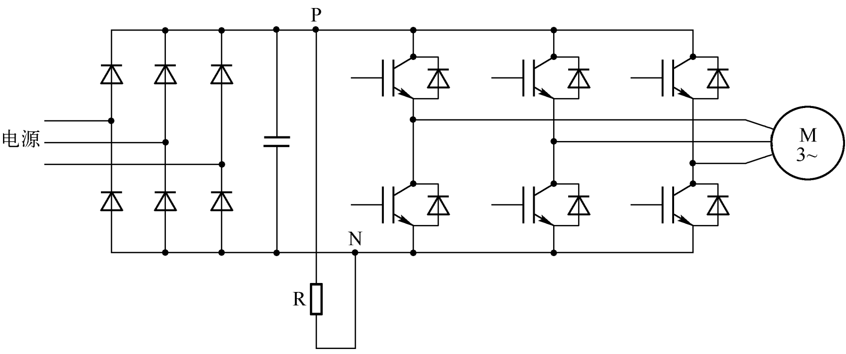


图 9-30　电阻制动原理图

【例 9-11】 MM 440 变频器的制动电阻如何接线？

【解】

MM 440 变频器用了外接电阻的制动方法，其连线如图 9-31 所示。

注意制动电阻 R 上的开关触点要与接触器 KM 的线圈串联，这样当制动电阻过热时，制动电阻上的热敏电阻切断接触器 KM 的电源，从而切断变频器的供电电源起到保护作用。

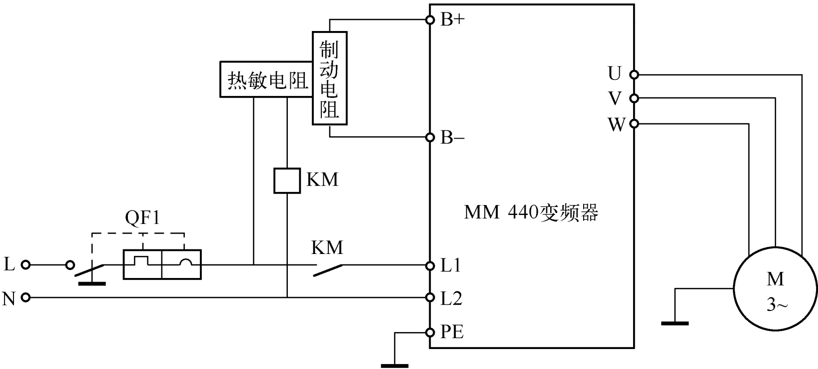


图 9-31　接线图

9.6.2 使用变频器时，电动机的起停控制

起停控制是最为常见的控制，以下用一个例子介绍。

【例 9-12】请设计变频器控制电动机起停的电路。

【解】

变频器的起停控制如图 9-32 所示，变频器以西门子 MM 440 为例，DIN1 实际是控制端子 5，+24 V 是端子 9。当 DIN1 和 +24 V 短接时，变频器起动。

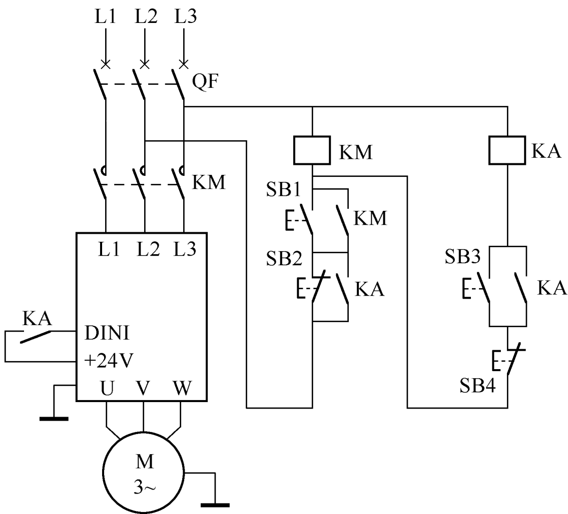


图 9-32 起动控制

1. 电路中各元器件的作用

① QF 断路器，主电源通断开关。

② KM 接触器，变频器通断开关。

③ SB1 按钮，变频器通电。

④ SB2 按钮，变频器断电。

⑤ SB3 按钮，变频器正转起动。

⑥ SB4 按钮，变频器停止。

⑦ KA 中间继电器，正转控制。

2. 控制过程

（1）变频器通断电的控制

当 SB 1 按下，KM 线圈通电，其触点吸合，变频器通电；按下 SB2，KM 线圈失电，触点断开，变频器断电。

（2）变频器起停的控制

按下 SB3，中间继电器 KA 线圈得电吸合，其触点将变频器的 DIN1 与 +24 V 短路，电动机正向转动。此时 KA 的另一常开触点封锁 SB2，使其不起作用，这就保证了变频器在正向转动期间不能使用电源开关进行停止操作。

当需要停止时，必须先按下 SB4，使 KA 线圈失电，其常开触点断开（电动机减速停

止)，这时才可按下 SB2，使变频器断电。

以上的电路，并没有将变频器的保护作用设计在电路中。

9.6.3 变频器的正反转控制

很多生产机械都要利用变频器的正反转控制，以下用一个例子来进行介绍。

【例 9-13】请设计变频器控制电动机正反转的电路。

【解】

变频器控制电动机正反转的电路如图 9-33 所示，变频器以西门子 MM 440 为例，DIN1 实际是控制端子 5，DIN2 实际是控制端子 6， +24 V 是端子 9。当 DIN1 和 +24 V 短接时，变频器正转；当 DIN2 与 +24 V 短接时，变频器反转。

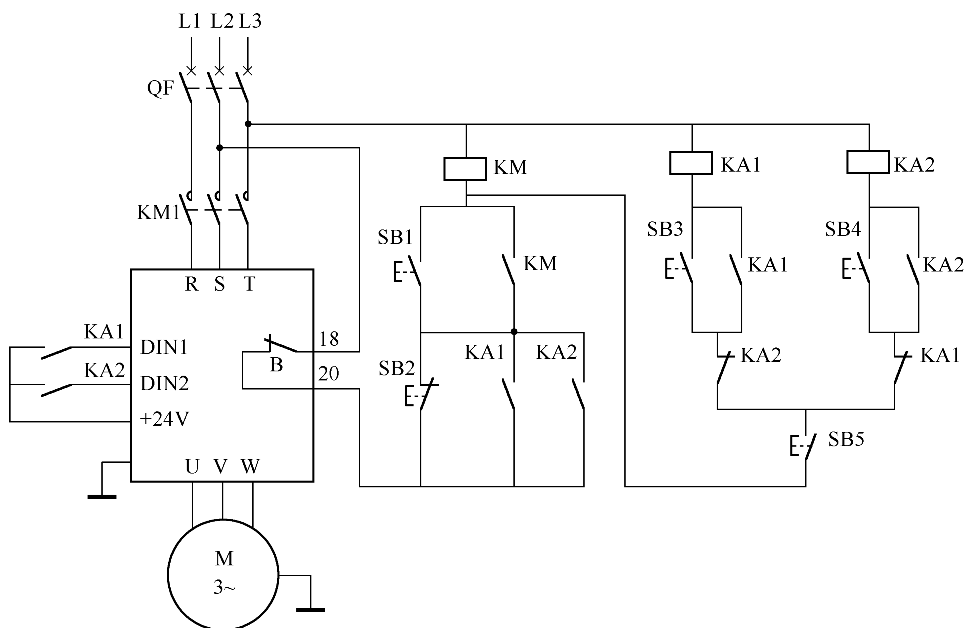


图 9-33 正反转控制电路图

1. 电路中各元器件的作用

① SB1 按钮，变频器通电。

② SB2 按钮，变频器断电。

③ SB3 按钮，正转起动。

④ SB4 按钮，反转起动。

⑤ SB5 按钮，电动机停止。

⑥ KA1 继电器，正转控制。

⑦ KA2 继电器，反转控制。

2. 电路在设计要点

1）KM 接触器仍只作为变频器的通、断电控制，而不作为变频器的运行与停止控制。因此，断电按钮 SB2 仍由继电器 KA1 或 KA2 封锁，运行时使 SB2 不起作用。

接触器 KM 的作用：变频器的保护功能动作时，可以通过接触器迅速切断电源。可以方

便地实现自锁、互锁控制。

2）控制电路串接报警输出接点 18、20，当变频器故障报警时切断控制电路，KM 断开而停机。

3）变频器的通、断电，正、反转运行控制均采用主令按钮。

4）正反转继电器 KA1 和 KA2 互锁，正反转切换不能直接进行，必须先停机再改变转向。

3. 变频器的正反转控制

1）正转。当按下 SB1，KM 线圈得电吸合，其主触点接通，变频器通电处于待机状态。与此同时，KM 的辅助常开触点使 SB1 自锁。这时如按下 SB3，KAl 线圈得电吸合，其常开触点 KA1 接通变频器的 DIN1 端子，电动机正转。与此同时，其另一常开触点闭合使 SB3 自锁，常闭触点断开，使 KA2 线圈不能通电。

2）反转。如果要使电动机反转，先按下 SB5 使电动机停止。然后按下 SB4，KA2 线圈得电吸合，其常开触点 KA2 闭合，接通变频器 REV 端子，电动机反转。与此同时，其另一常开触点 KA2 闭合使 SB4 自保，常闭触点 KA2 断开，使 KAl 线圈不能通电。

3）停止。当需要断电时，必须先按下 SB5，使 KA1 和 KA2 线圈失电，其常开触点断开（电动机减速停止），并解除对 SB2 的旁路，这时才能可按下 SB2，使变频器断电。变频器故障报警时，控制电路被切断，变频器主电路断电。

4）控制电路的特点：

① 自锁保持电路状态的持续，KM 自锁，持续通电；KA1 自锁，持续正转；KA2 自锁，持续反转。

② 互锁保持变频器状态的平稳过渡，避免变频器受冲击。KA1、KA2 互锁，正、反转运行不能直接切换；KA1、KA2 对 SB2 的锁定，保证运行过程中不能直接断电停机。

③ 主电路的通断由控制电路控制，操作更安全可靠。

4. 参数设置

变频器的参数设置见表 9-17。表格中电动机的参数，如额定电压、额定电流都应根据实际情况而定。

表 9-17　变频器参数

序号	变频器参数	出厂值	设定值	功 能 说 明
1	P0304	230	380	额定电压 380 V
2	P0305	3. 25	0. 35	额定电流 0. 35 A
3	P0307	0. 75	0. 06	额定功率 0. 06 kW
4	P0310	50	50	额定频率 50 Hz
5	P0311	0	1 440	额定转速 1 440 r/min
6	P0700	2	2	选择命令源
7	P1000	2	1	频率源
8	P0701	1	1	正转
9	P0702	12	2	反转

【例9–14】 变频器的通断电是在停止输出状态下进行的，为什么在运行状态下一般不允许切断电源？

【解】

1）变频器内部电路的原因。突然断电对主电路安全工作不利。

2）负载的原因。电源突然断电，变频器立即停止输出，运转中的电动机处于自由停止状态，这对于某些运行场合也会造成影响。

小　　结

（1）掌握变频器的参数设定是正确使用变频器的前提。

（2）理解变频器的“交－直－交”工作原理。

（3）掌握变频器BOP速度给定、多段速度给定、模拟量速度给定和通信速度给定的应用场合以及其在运输站上的应用。

（4）掌握PLC控制变频器速度给定的接线方法，特别注意当PLC为晶体管输出时，若PLC为PNP输出，则要将变频器的输入调整为PNP输入，同理若PLC为NPN输出，则要将变频器的输入调整为NPN输入。

（5）通信速度给定的难点是理解各个控制字的含义，这是非常关键的，此外对变频器参数的正确设定也十分关键。

习　　题

1. 简述变频器的“交－直－交”工作原理。
2. 三相交流异步电动机有几种调速方式？
3. 使用变频器时，一般有几种速度给定方式？
4. 变频器电源输入端接到电源输出端后，有什么后果？
5. 使用变频器时，制动原理是什么？
6. 使用变频器时，电动机的正反转怎样实现？
7. 例9–7中，若将PLC改为继电器输出，则应该怎样接线？
8. 例9–7中，若将PLC改为三菱的FX2N－48MT，则应该怎样接线？变频器的参数设置有和变化？（提示：请参考三菱FX系统手册和西门子变频器使用大全）
9. 不用西门子的指令库，也能建立两台CPU 226CN的Modbus通信，这句话对吗？为什么？
10. 不用西门子的指令库，也能建立两台CPU 226CN和MM 440变频器的USS通信，这句话对吗？
11. 是否所有型号的S7－200都能和MM 440变频器建立USS通信？

第 10 章　S7 - 200 系列 PLC 的其他应用技术

本章介绍 PID 的基本原理、PID 在电炉温度控制中的应用，以及程序的各种下载方法。

10.1　S7 - 200 PLC 在 PID 中的应用

10.1.1　PID 控制原理简介

在过程控制中，按偏差的比例（P）、积分（I）和微分（D）进行控制的 PID 控制器（也称 PID 调节器）是应用最广泛的一种自动控制器。它具有原理简单，易于实现，适用面广，控制参数相互独立，参数选定比较简单，调整方便等优点；而且在理论上可以证明，对于过程控制的典型对象——“一阶滞后 + 纯滞后”与“二阶滞后 + 纯滞后”的控制对象，PID 控制器是一种最优控制。PID 调节规律是连续系统动态品质校正的一种有效方法，它的参数整定方式简便，结构改变灵活（如可为 PI 调节、PD 调节等）。长期以来，PID 控制器被广大科技人员及现场操作人员所采用，并积累了大量的经验。

PID 控制器就是根据系统的误差，利用比例、积分、微分计算出控制量来进行控制。当被控对象的结构和参数不能完全掌握，或得不到精确的数学模型、或控制理论的其他技术难以采用时，系统控制器的结构和参数必须依靠经验和现场调试来确定，这时应用 PID 控制技术最为方便。即当我们不完全了解一个系统和被控对象，或不能通过有效的测量手段来获得系统参数时，最适合采用 PID 控制技术。

1. 比例（P）控制

比例控制是一种最简单、最常用的控制方式，如放大器、减速器和弹簧等。比例控制器能立即成比例地响应输入的变化量。但仅有比例控制时，系统输出存在稳态误差（Steady - state error）。

2. 积分（I）控制

在积分控制中，控制器的输出量是输入量对时间的积累。对一个自动控制系统，如果在进入稳态后存在稳态误差，则称这个控制系统是有稳态误差的或简称有差系统（System with Steady - state error）。为了消除稳态误差，在控制器中必须引入“积分项”。积分项对误差的运算取决于时间的积分，随着时间的增加，积分项会增大。所以即便误差很小，积分项也会随着时间的增加而加大，它推动控制器的输出增大，使稳态误差进一步减小，直到等于零。因此，采用比例 + 积分（PI）控制器，可以使系统在进入稳态后无稳态误差。

3. 微分（D）控制

在微分控制中，控制器的输出与输入误差信号的微分（即误差的变化率）成正比关系。自动控制系统在克服误差的调节过程中可能会出现振荡甚至失稳。其原因是由于存在有较大的惯性组件（环节）或有滞后（delay）组件，具有抑制误差的作用，其变化总是落后于误

差的变化。解决的办法是使抑制误差的作用的变化“超前”，即在误差接近零时，抑制误差的作用就应该是零。这就是说，在控制器中仅引入“比例”项往往是不够的，比例项的作用仅是放大误差的幅值，而目前需要增加的是“微分项”，它能预测误差变化的趋势，这样，具有比例+微分（PD）的控制器就能够提前使抑制误差的控制作用等于零，甚至为负值，从而避免被控量的严重超调。所以对有较大惯性或滞后的被控对象，比例+微分（PD）控制器能改善系统在调节过程中的动态特性。

4. 闭环控制系统特点

控制系统一般包括开环控制系统和闭环控制系统。开环控制系统（Open－loop Control System）是指被控对象的输出（被控制量）对控制器（controller）的输出没有影响，在这种控制系统中，不依赖将被控制量反送回来以形成任何闭环回路。闭环控制系统（Closed－loop Control System）的特点是系统被控对象的输出（被控制量）会反送回来影响控制器的输出，形成一个或多个闭环。闭环控制系统有正反馈和负反馈，若反馈信号与系统给定值信号相反，则称为负反馈（Negative Feedback）；若极性相同，则称为正反馈。一般闭环控制系统均采用负反馈，又称负反馈控制系统。可见，闭环控制系统性能远优于开环控制系统。

5. PID 控制器的参数整定

PID 控制器的参数整定是控制系统设计的核心内容。它是根据被控过程的特性，确定 PID 控制器的比例系数、积分时间和微分时间的大小。PID 控制器参数整定的方法很多，概括起来有如下两大类：

① 理论计算整定法。它主要依据系统的数学模型，经过理论计算确定控制器参数。这种方法所得到的计算数据未必可以直接使用，还必须通过工程实际进行调整和修改。

② 工程整定法。它主要依赖于工程经验，直接在控制系统的试验中进行，且方法简单、易于掌握，在工程实际中被广泛采用。PID 控制器参数的工程整定方法，主要有临界比例法、反应曲线法和衰减法。这三种方法各有其特点，其共同点都是通过试验，然后按照工程经验公式对控制器参数进行整定。但无论采用哪一种方法所得到的控制器参数，都需要在实际运行中进行最后的调整与完善。

现在一般采用的是临界比例法。利用该方法进行 PID 控制器参数的整定步骤如下：

① 首先预选择一个足够短的采样周期让系统工作。

② 仅加入比例控制环节，直到系统对输入的阶跃响应出现临界振荡，记下这时的比例放大系数和临界振荡周期。

③ 在一定的控制度下通过公式计算得到 PID 控制器的参数。

6. PID 控制器的主要优点

PID 控制器成为应用最广泛的控制器，它具有以下优点：

1）PID 算法蕴涵了动态控制过程中过去、现在、将来的主要信息，而且其配置几乎最优。其中，比例（P）代表了当前的信息，起纠正偏差的作用，使过程反应迅速。微分（D）在信号变化时有超前控制作用，代表将来的信息。在过程开始时强迫过程进行，过程结束时减小超调，克服振荡，提高系统的稳定性，加快系统的过渡过程。积分（I）代表了过去积累的信息，它能消除静差，改善系统的静态特性。此三种作用配合得当，可使动态过程快速、平稳、准确，收到良好的效果。

2）PID 控制适应性好，有较强的鲁棒性，对各种工业应用场合，都可在不同的程度上

应用。特别适于“一阶惯性环节+纯滞后”和“二阶惯性环节+纯滞后”的过程控制对象。

3）PID 算法简单明了，各个控制参数相对较为独立，参数的选定较为简单，形成了完整的设计和参数调整方法，很容易为工程技术人员所掌握。

4）PID 控制根据不同的要求，针对自身的缺陷进行了不少改进，形成了一系列改进的 PID 算法。例如，为了克服微分带来的高频干扰的滤波 PID 控制，为克服大偏差时出现饱和超调的 PID 积分分离控制，为补偿控制对象非线性因素的可变增益 PID 控制等。这些改进算法在一些应用场合取得了很好的效果。同时当今智能控制理论的发展，又形成了许多智能 PID 控制方法。

7. PID 的算法

PID 控制器调节输出，保证偏差（e）为零，使系统达到稳定状态，偏差是给定值（SP）和过程变量（PV）的差。PID 控制的原理基于以下公式：

$$M(t) = K_C \cdot e + K_C \cdot \int_0^1 e\mathrm{d}t + M_{\text{initial}} + K_C \cdot \frac{\mathrm{d}e}{\mathrm{d}t} \tag{10-1}$$

式中，$M(t)$是 PID 回路的输出；K_C 是 PID 回路的增益；e 是 PID 回路的偏差（给定值与过程变量的差）；M_{initial}是 PID 回路输出的初始值。

由于以上的算式是连续量，必须将连续量离散化才能在计算机中运算，离散处理后的算式如下：

$$M_n = K_C \cdot e_n + K_I \cdot \sum_1^n e_x + M_{\text{initial}} + K_D \cdot (e_n - e_{n-1}) \tag{10-2}$$

式中，M_n 是在采样时刻 n，PID 回路输出的计算值；K_C 是 PID 回路的增益；K_I 是积分项的比例常数；K_D 是微分项的比例常数；e_n 是采样时刻 n 回路的偏差值；e_{n-1}是采样时刻 $n-1$ 回路的偏差值；e_x 是采样时刻 x 回路的偏差值；M_{initial}是 PID 回路输出的初始值。

再对以上算式进行改进和简化，得出如下计算 PID 输出的算式：

$$M_n = MP_n + MI_n + MD_n \tag{10-3}$$

式中，M_n 是第 n 采样时刻的计算值；MP_n 是第 n 采样时刻的比例项值；MI_n是第 n 采样时刻的积分项的值；MD_n 是第 n 采样时刻微分项的值。

$$MP_n = K_C \cdot (SP_n - PV_n) \tag{10-4}$$

式中，MP_n 是第 n 采样时刻的比例项值；K_C 是增益；SP_n 是第 n 次采样时刻的给定值；PV_n 是第 n 次采样时刻的过程变量值。很明显，比例项 MP_n 数值的大小和增益 K_C 成正比，增益 K_C 增加可以直接导致比例项 MP_n 的快速增加，从而直接导致 M_n 增加。

$$MI_n = K_C \cdot T_S/T_I \cdot (SP_n - PV_n) + MX \tag{10-5}$$

式中，K_C 是增益；T_S 是回路的采样时间；T_I 是积分时间；SP_n 是第 n 次采样时刻的给定值；PV_n 是第 n 次采样时刻的过程变量值；MX 是第 $n-1$ 时刻的积分项（也称为积分前项）。很明显，积分项 MI_n 数值的大小随着积分时间 T_I 的减小而增加，T_I 的减小可以直接导致积分项 MI_n 数值的增加，从而直接导致 M_n 增加。

$$MD_n = K_C \cdot (PV_{n-1} - PV_n) \cdot T_D/T_S \tag{10-6}$$

式中，K_C 是增益；T_S 是回路的采样时间；T_D 是微分时间；PV_n 是第 n 次采样时刻的过程变量值；PV_{n-1}是第 $n-1$ 次采样时刻的过程变量。很明显，微分项 MD_n 数值的大小随着微分时间 T_D 的增加而增加，T_D 的增加可以直接导致积分项 MD_n 数值的增加，从而直接导

致 M_n 增加。

【关键点】 式（10-3）～式（10-6）是非常重要的。根据这几个公式，读者必须建立一个概念：增益 K_C 增加可以直接导致比例项 MP_n 的快速增加，T_I 的减小可以直接导致积分项 MI_n 数值的增加，微分项 MD_n 数值的大小随着微分时间 T_D 的增加而增加，从而直接导致 M_n 增加。理解了这一点，对于正确调节 P、I、D 三个参数是至关重要的。

10.1.2 利用 S7-200 PLC 进行电炉的温度控制

【例 10-1】 有一台电炉，要求炉温控制在一定的范围。电炉的工作原理如下：

当设定电炉温度后，S7-200 经过 PID 运算后由模拟量输出模块 EM 232 输出一个电压信号送到控制板，控制板根据电压信号（弱电信号）的大小控制电热丝的加热电压（强电）的大小（甚至断开），温度传感器测量电炉的温度，温度信号经过控制板的处理后输入到模拟量输入模块 EM 231，再送到 S7-200 进行 PID 运算，如此循环。整个系统的硬件配置如图 10-1 所示。

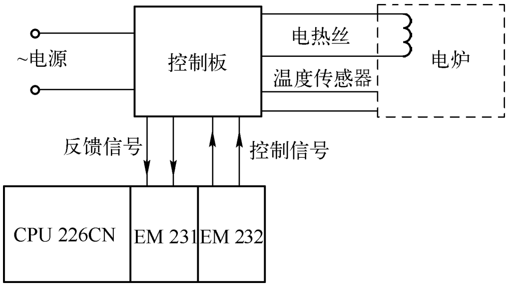


图 10-1　硬件配置图

【解】

1. 主要软硬件配置

① 1 套 STEP 7-Micro/WIN V4.0。

② 1 台 CPU 226CN。

③ 1 台 EM 231。

④ 1 台 EM 232。

⑤ 1 根编程电缆（或者 CP 5611 卡）。

⑥ 1 台电炉（含控制板）。

2. 主要指令介绍

PID 回路（PID）指令，当使能有效时，根据表格（TBL）中的输入和配置信息指定回路执行 PID 计算。PID 指令的格式见表 10-1。

PID 指令使用注意事项：

① 程序中最多可以使用 8 条 PID 指令，回路号为 0～7，不能重复使用。

② 必须保证过程变量和给定值积分项前值和过程变量前值在 0.0～1.0 之间。

③ 如果进行 PID 计算的数学运算时遇到错误，将设置 SM1.1（溢出或非法数值）并终止 PID 指令的执行。

表 10-1　PID 指令格式

LAD	输入/输出	含　　义	数 据 类 型
PID EN　ENO TBL LOOP	EN	使能	BOOL
	TBL	参数表的起始地址	BYTE
	LOOP	回路号，常数范围 0 ~ 7	BYTE

在工业生产过程中，模拟信号 PID（由比例、积分和微分构成的闭合回路）调节是常见的控制方法。运行 PID 控制指令，S7－200 将根据参数表中输入测量值、控制设定值及 PID 参数，进行 PID 运算，求得输出控制值。参数表中有 9 个参数，共占用 36 个字节，全部是 32 位的实数，部分保留给自整定用。PID 控制回路的参数表见表 10-2。

表 10-2　PID 控制回路参数表

偏移地址	参　　数	数据格式	参 数 类 型	描　　述
0	过程变量 PVn	REAL	输入/输出	必须在 0.0 ~ 1.0 之间
4	给定值 SPn	REAL	输入	必须在 0.0 ~ 1.0 之间
8	输出值 Mn	REAL	输入	必须在 0.0 ~ 1.0 之间
12	增益 K_c	REAL	输入	增益是比例常数，可正可负
16	采样时间 T_s	REAL	输入	单位为秒，必须是正数
20	积分时间 T_I	REAL	输入	单位为分钟，必须是正数
24	微分时间 T_d	REAL	输入	单位为分钟，必须是正数
28	上一次积分值 M_X	REAL	输入/输出	必须在 0.0 ~ 1.0 之间
32	上一次过程变量 PV_{n-1}	REAL	输入/输出	最后一次 PID 运算过程变量值
36 ~ 79	保留自整定变量			

3. I/O 分配

在 I/O 分配之前，先计算所需要的 I/O 点数，输入点为 3 个，输出点为 1 个，由于输入输出最好留 15% 左右的余量备用，所用初步选择的 PLC 是 CPU 222CN 或者 CPU 221CN，但 CPU 221CN 不能带扩展模块，所以选择 CPU 222CN 模块。又因为控制对象为接触器，所以 PLC 最后定为 CPU 222CN（AC/DC/继电器）。电炉机控制系统的 I/O 分配表见表 10-3。

表 10-3　I/O 分配表

输　　入			输　　出		
名称	符号	输入点	名称	符号	输出点
起动按钮	SB1	I0.0	接触器	KM1	Q0.0
停止按钮	SB2	I0.1			
急停按钮	SB3	I0.2			

4. 设计电气原理图

根据 I/O 分配表和题意，其设计原理图如图 10-2 所示。

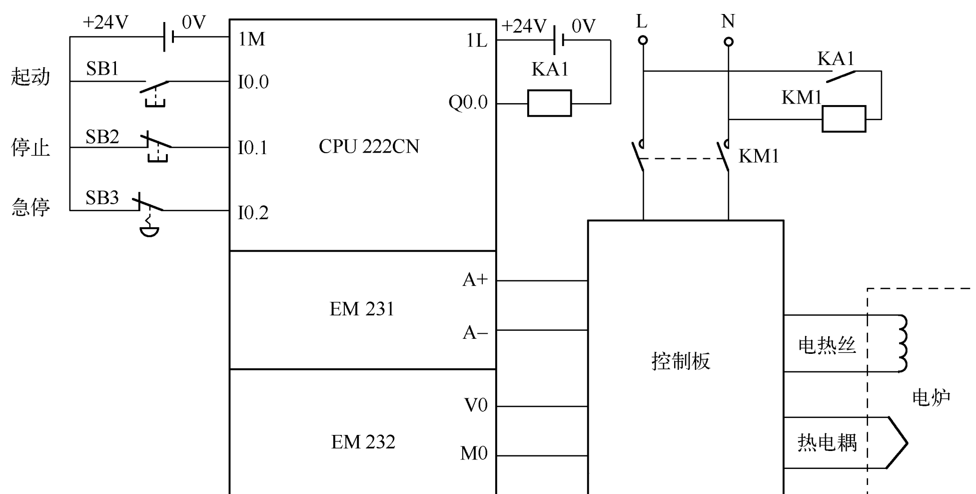


图 10–2　原理图

5. 编写程序

再编写 PLC 控制程序，共有三种方案，第一种方案不需要使用 PID 指令，直接从底层算法编写程序，这种编程方法，PID 的控制精度较高，但对编写者水平要求高，本书不作介绍，读者可以参考相关文献；第二种方法采用指令向导，较常用，后续将介绍；第三种方法直接用 PID 指令编写程序，其程序如图 10–3 所示。

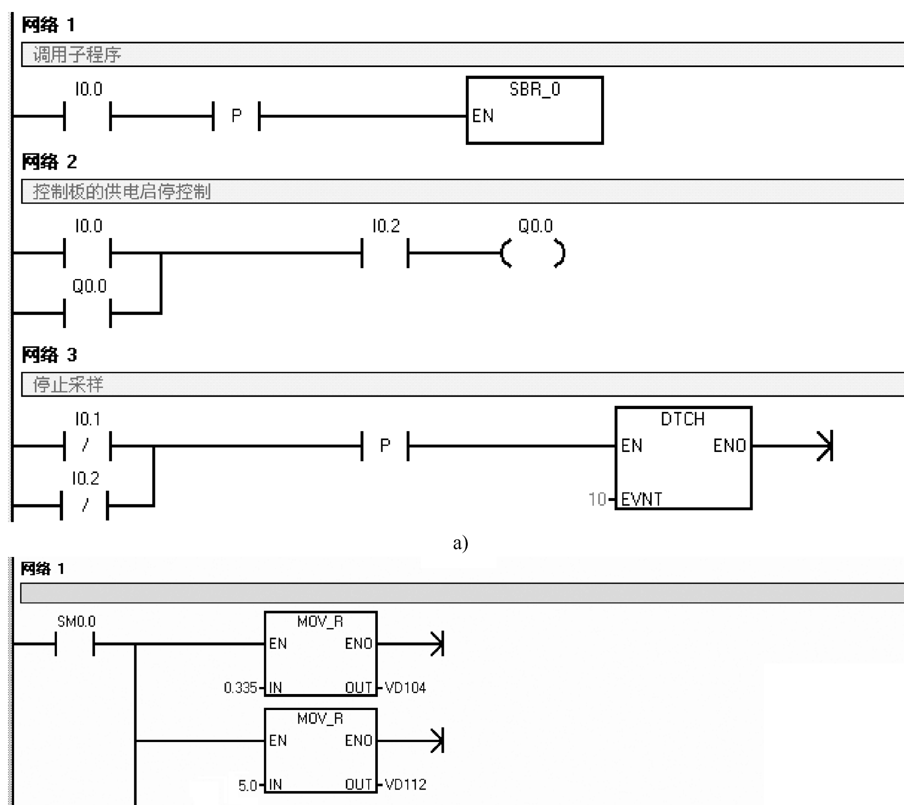
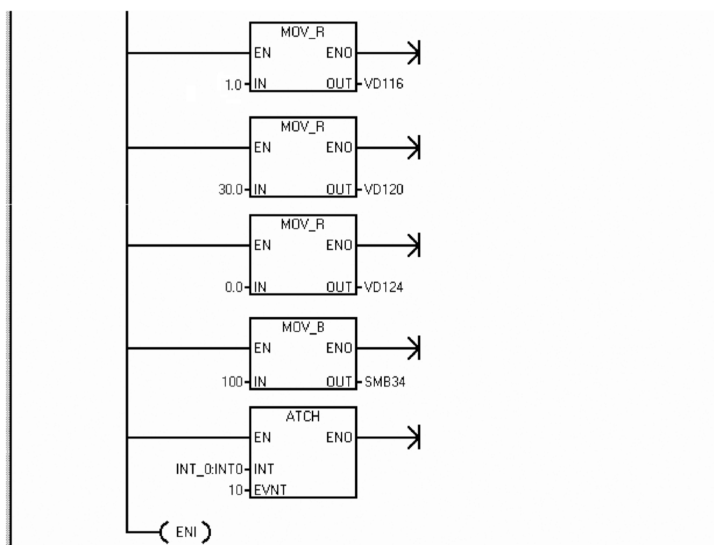
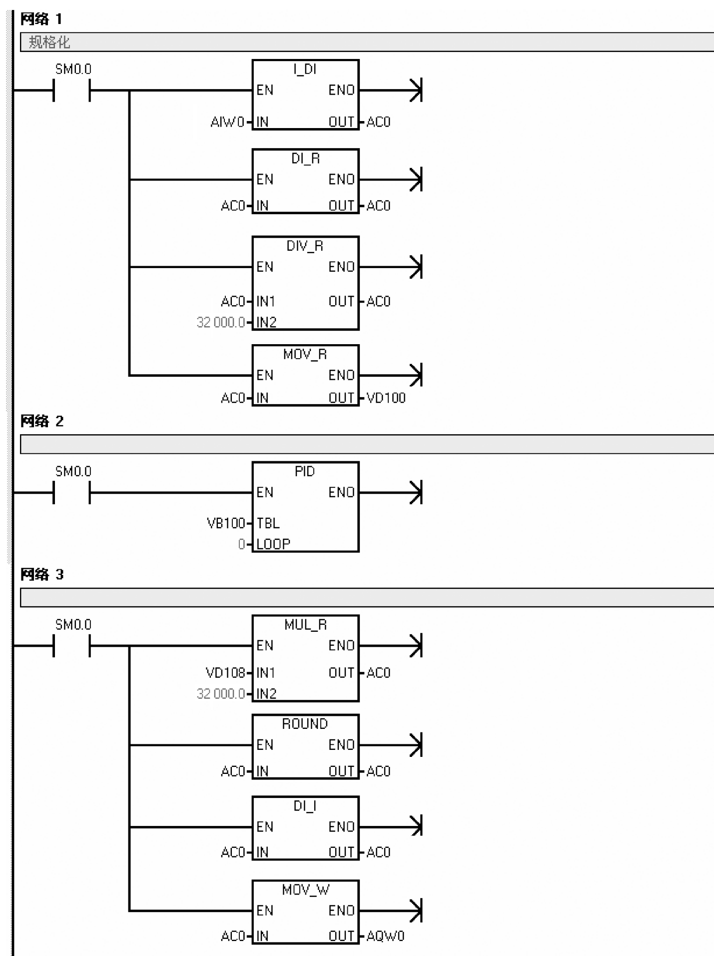


图 10–3　PID 控制程序



b)



c)

图 10-3　电炉 PID 控制程序（续）

【关键点】编写此程序首先要理解 PID 的参数表各个参数的含义，其次是要理解数据类型的转换。要将整数转化成实数，必须先将整数转化成双整数，因为 S7－200 中没有直接将整数转化成实数的指令。

用指令向导生成子程序进行 PID 控制的方法比较常用，以下进行详细介绍。

（1）用指令向导生成子程序

1）打开指令向导。单击菜单栏中的“工具”→“指令向导”，即可打开指令向导画面，如图 10-4 所示。或者单击左侧“工具”→“指令向导”，也可以打开指令向导画面。



图 10-4　打开指令向导

2）选择 PID 选项。指令向导有三个选项，即 PID、网络读写和高速计数器，选择“PID”选项，单击“下一步”按钮，如图 10-5 所示。

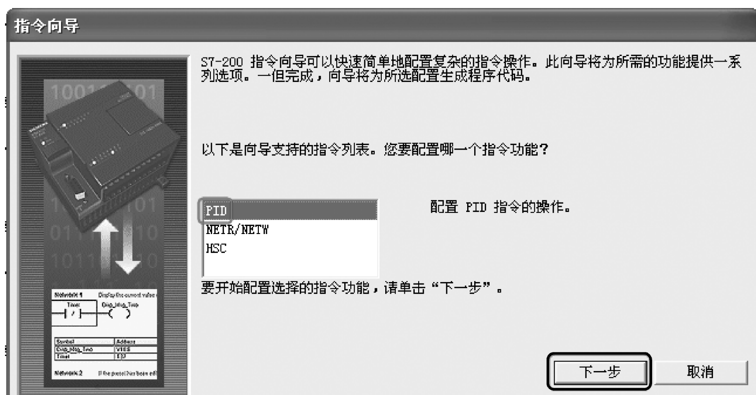


图 10-5　选择 PID 选项

3）指定回路号码。S7－200 最多允许 8 个回路，当只有一个回路时，可选择默认的回路号为“0”，如图 10-6 所示，单击“下一步”按钮。

4）设置回路参数。如图 10-7 所示，比例增益就是比例环节的参数（即 P），本例设置为“2”；采样时间是 0.2 s，如果参数的变化快于 0.2 s 将不能采样，S7－200 规定最小采样时间是 0.1 s；本例积分时间是 10 min（即 I），如果要取消积分环节，则在积分事件中填入“INF”（即无穷大）；本例的微分时间为 0 min（即 D），也就是微分环节被取消。给定的范围本例为 0.0～100.0，可以理解为温度范围是 0～100℃（如果实际温度是－10～1000℃，则按照实际填入），也可以理解为范围是 0～100%。以上参数可在调试时修改。单击“下一

步”按钮。

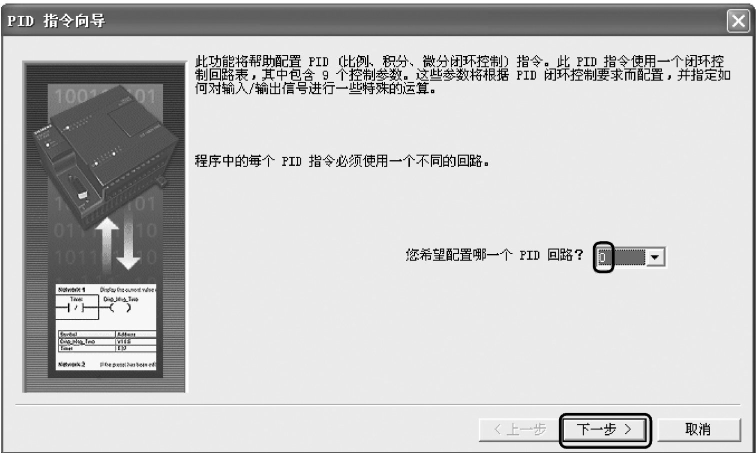


图 10-6　指定回路号码

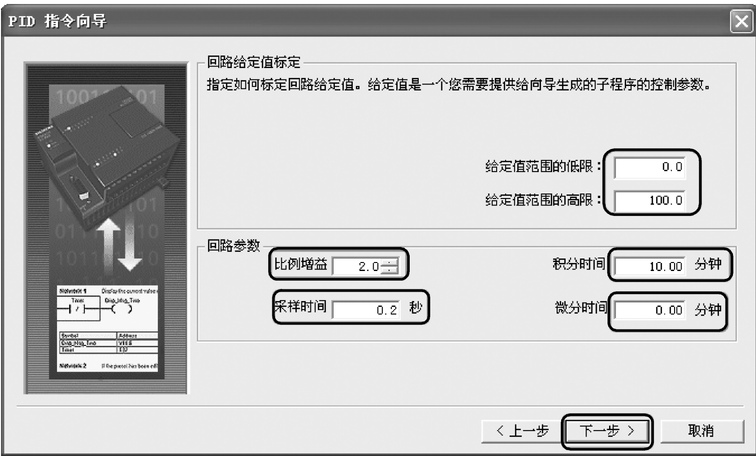


图 10-7　设置回路参数

5）设置回路输入和输出选项。分别如下：

① 回路输入选项。“标定”中有单极性和双极性 2 个选项，代表输入信号的极性（过零则为双极性，如 ±2 V，不过零为单极性），如果输入信号是 4 ~ 20 mA 的信号，可选择“使用 20% 偏移量”；“过程变量”和回路给定值有一个对应关系，过程变量 0，对应回路给定值 0℃，过程变量 32 000，对应回路给定值 100℃。如图 10-8 所示。

② 回路输出选项。“输出类型”有模拟量和数字量，2 个选项，本例使用的是 EM 232 模拟量模块，所以选择“模拟量”；“标定”中有单极性和双极性 2 个选项，代表输出信号的极性（过零则为双极性，如 ±2 V，不过零为单极性），如果输出信号是 4 ~ 20 mA，可选择“使用 20% 偏移量”；范围低限和范围高限是 D/A 转换的数字量的范围，选择默认值。最后单击“下一步”按钮。

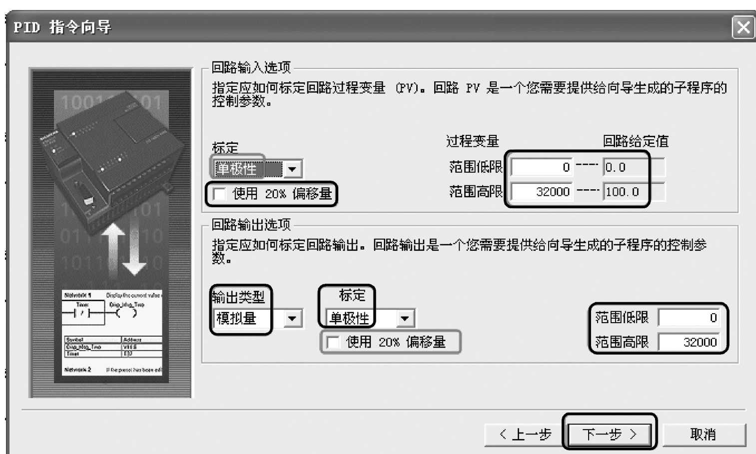


图 10-8　设置回路输入和输出选项

6）设置回路报警选项。当达到报警条件时，输出被置位，产生报警。如图 10-9 所示，当温度低于 100℃（10%）时报警，当温度高于 90℃（90%）时也报警。单击“下一步”按钮。



图 10-9　设置回路报警选项

7）为计算指定存储区。PID 指令使用 V 存储区中的一个 36 个字节的参数表，存储用于控制回路操作的参数。这个 V 存储区可以读者分配也可以使用系统默认 V 存储区，但要注意，这个 V 存储区被系统占用后，读者编程时，不可以再使用，否则可能导致错误的结果。如图 10-10 所示，单击“下一步”按钮。

8）指定子程序和中断程序。如图 10-11 所示，为子程序和中断程序的名称，这两个名称读者可以修改，也可以使用系统默认的名称，由于有多个回路时，初始化子程序不同，但多个回路使用同一个中断程序（如 PID_EXE），修改名称容易出错，所以作者不建议修改名称。如果需要手动控制，则勾选“增加 PID 手动控制”选项。单击“下一步”按钮。



图 10-10　为计算指定存储区

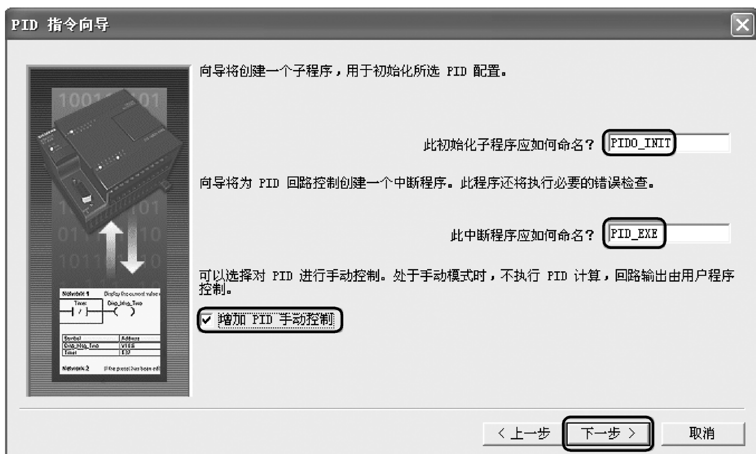


图 10-11　指定子程序和中断程序

【**关键点**】本例的中断程序 PID_EXE，使用了定时中断 0，所以读者若还要使用定时中断，则只能使用定时中断 1。

9）生成 PID 代码。如图 10-12 所示，单击“完成”按钮，即可生成 PID 代码。

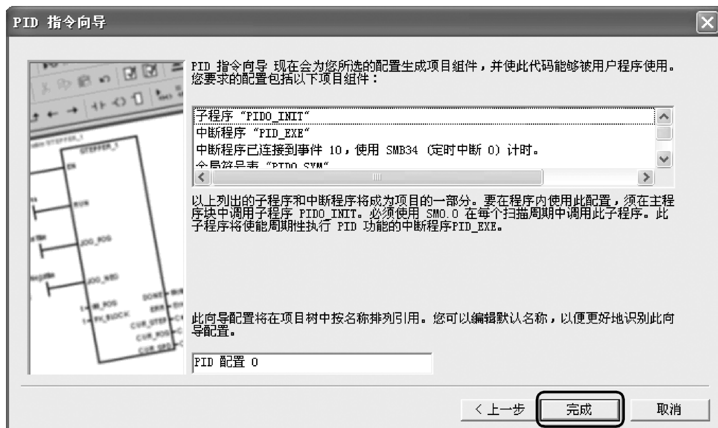
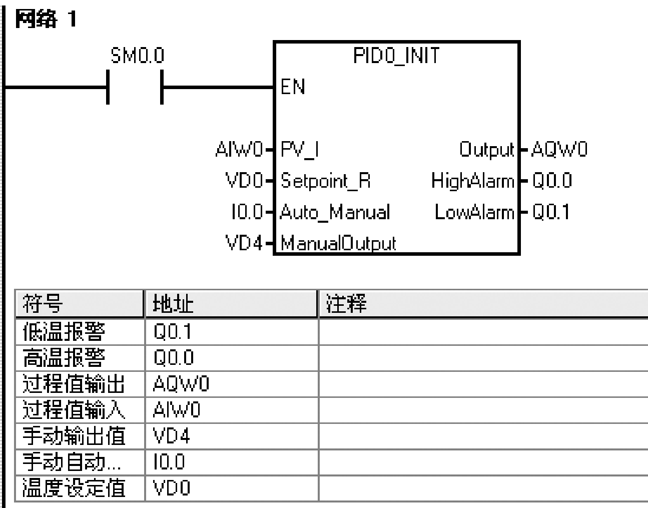


图 10-12　生成 PID 代码

（2）编写和下载程序

程序和符号表如图 10-13 所示，并将程序下载到 PLC 中。



符号	地址	注释
低温报警	Q0.1	
高温报警	Q0.0	
过程值输出	AQW0	
过程值输入	AIW0	
手动输出值	VD4	
手动自动...	I0.0	
温度设定值	VD0	

图 10-13　程序

（3）调试。

1）打开 PID 调节控制面板。单击“工具”→“PID 调节控制面板”，PID 调节控制面板如图 10-14 所示。PID 调节控制面板只能在使用指令向导生成的 PID 程序中使用。

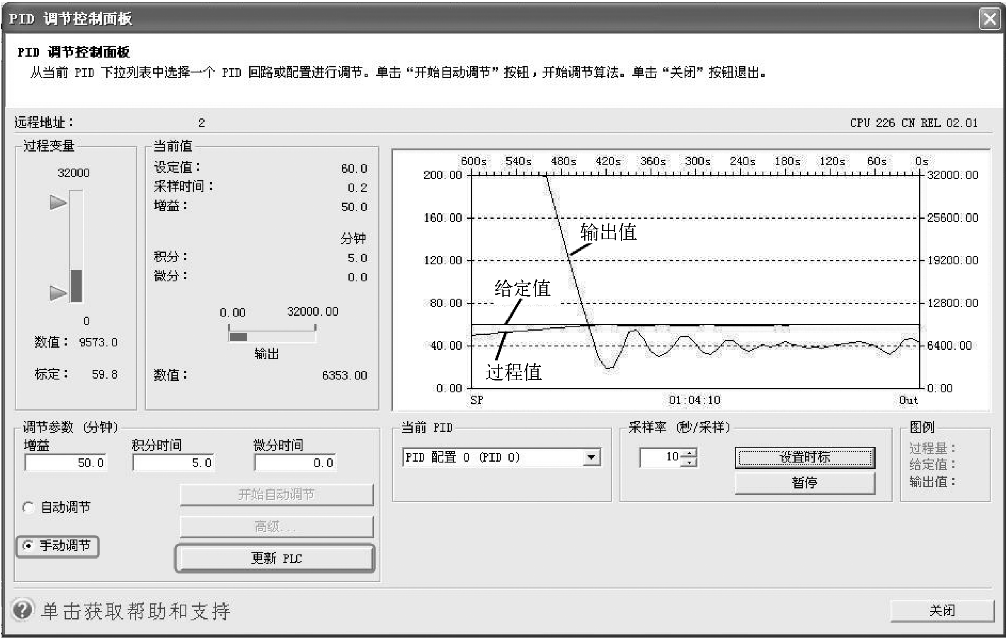


图 10-14　PID 调节控制面板（1）

从图 10-14 可以看出当过程值（传感器测量数值）比给定值（即设定值，本例为 60.0℃）相差较大时，输出值很大，随着过程值接近给定值时，输出值减小，并在一定的

范围内波动。

2）选择手动调节模式。在“增益”（P）、“积分时间”（I）和“微分时间”（D）中输入参数，并单击“更新 PLC 参数”按钮，这样 P、I、D 三个参数就写入到 PLC 中。调试完成后可以看到如图 10－15 所示的画面，过程值和给定值基本重合（此图的给定值为 33.5℃）。

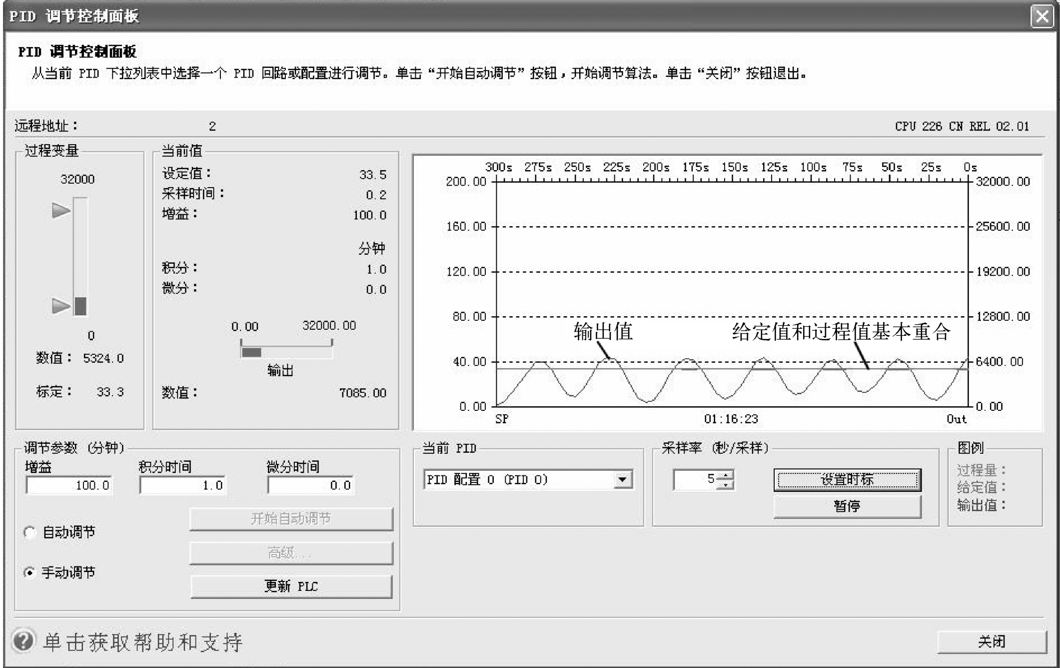


图 10-15　PID 调节控制面板（2）

10.2　程序的下载方法

多数西门子 S7－200 系列 PLC 的用户用编程电缆（适配器）下载程序（用 PC/PPI 电缆下载程序），但这并不意味着西门子 PLC 只能用这种方式下载程序。其实，西门子 PLC 的下载程序的方法比较灵活，下面以 PPI、PROFIBUS、MPI 和 TCP/IP 通信协议为例介绍程序的下载方法。

10.2.1　用 PPI 协议下载程序

采用 PPI 通信协议下载程序是较多采用的下载方法，常用的有两种方法，一是用 PC/PPI 编程电缆，另一种方法是用板卡（对于台式电脑用 CP 5611、CP 5613 卡，对于笔记本电脑用 CP 5511 卡）。

1. 用 PC/PPI 编程电缆下载程序

1）打开 PG/PC 接口。在图面的左侧的组件“工具”中选中“设置 PG/PC 接口”，并单击之，如图 10-16 所示。也可以在菜单中打开设置 PG/PC 接口界面，其路径为“查看”→“组件”→“设置 PG/PC 接口”。

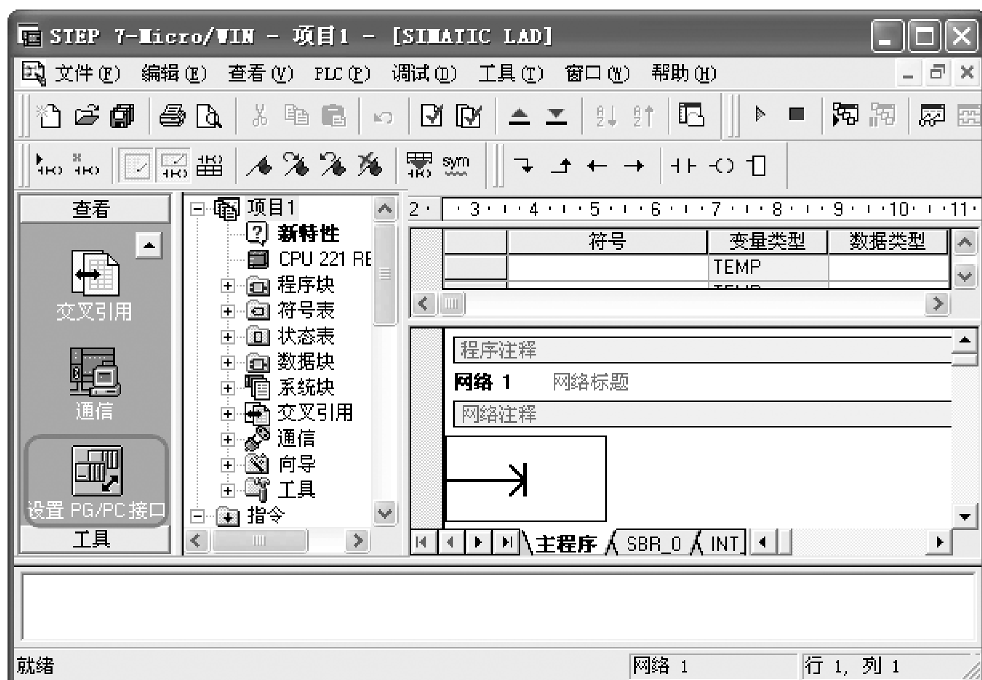


图 10-16　打开 PG/PC 接口

2）设置 PG/PC 接口。选中“PC/PPI cable（PPI）”，并单击“属性”按钮，如图 10-17 所示。

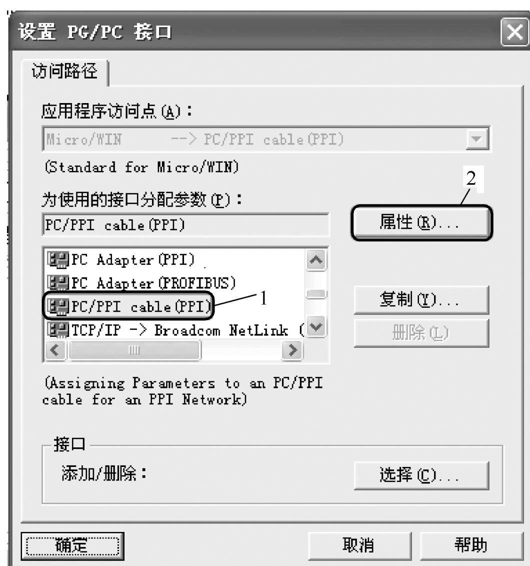


图 10-17　设置 PG/PC 接口

3）设置 PC 的通信接口。在“本地连接”选项卡中设置通信接口，PPI 电缆与计算机相连的接头有 RS-232C 和 USB 两种接口形式，按照实际选用的 PPI 电缆选择，本例使用的 PPI 电缆电脑连接端的接口为 USB，因此只能选择 USB，如图 10-18 所示。单击“确定”按

钮，回到如图 10-17 所示的界面。当然若读者使用的 PPI 电缆电脑连接端的接口为 RS-232C，也可以用 USB-RS232C，将 RS-232C 转化成 USB 接口形式。

4）设置通信参数。在“PPI”选项卡中设置通信参数，通常不用修改参数，先单击“默认”按钮，再单击“确定”按钮即可，如图 10-19 所示。

【关键点】有的 PPI 电缆上有波特率的设定拨钮，PPI 电缆上的波特率设定值必须与图 10-19 中的“传输率”的设定值相等，否则是不能建立通信的。本例的波特率设定值为9.6 kbit/s。



图 10-18 设置 PC 的通信接口



图 10-19 设置通信参数

5）打开通信界面。在图面的左侧的组件“工具”中选中“通信”，并单击之，如图 10-20所示。也可以在菜单中打开通信界面，其路径为“查看”→“组件”→“通信”。

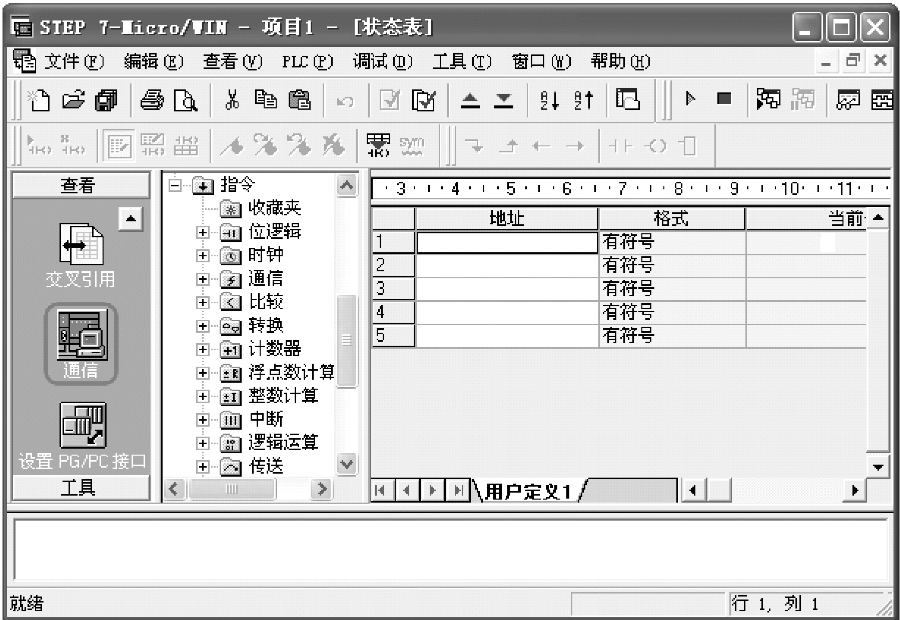


图 10-20 打开通信界面

6）建立 STEP 7 - Micro/WIN 与 PLC 的通信。双击“双击刷新”，如图 10-21 所示，STEP 7 - Micro/WIN 自动建立与 PLC 的通信连接，当两者通信建立成功时，弹出 PLC 的型号版本、地址和波特率信息，如图 10-22 的“1”处，再单击“确认”按钮即可。



图 10-21　建立通信（1）



图 10-22　建立通信（2）

7）下载程序。单击“下载”按钮，如图 10-23 所示，弹出下载页如图 10-24 所示，通常在选项中选择“程序块”、“数据块”和“系统块”三个选项，再单击“下载”按钮即可。正在下载程序的界面，如图 10-25 所示。

【关键点】 *若有配方和数据记录还要选择这两个选项，使用配方和数据记录时，PLC 必须配置存储卡。*

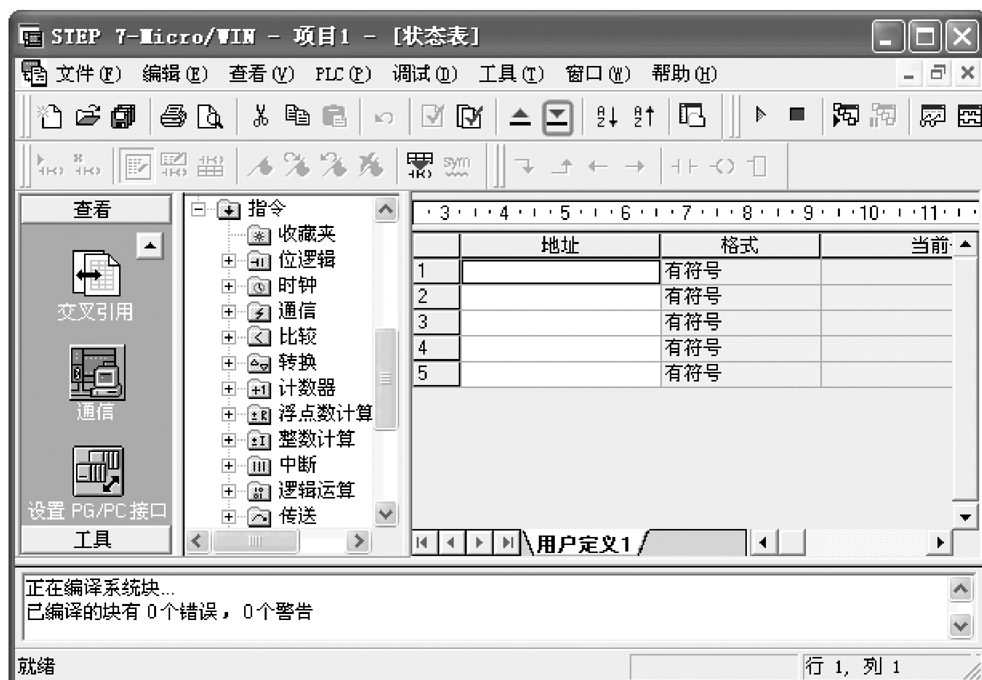


图 10-23　下载（1）



图 10-24　下载（2）



图 10-25　下载进行中

2. 用 CP 5611 卡下载程序

1）打开 PG/PC 接口。在图面的左侧的组件“工具”中选中“设置 PG/PC 接口”，并单击之，如图 10-26 所示。也可以在菜单中打开设置 PG/PC 接口界面，其路径为“查看”→“组件”→“设置 PG/PC 接口”。



图 10-26　设置 PG/PC 接口

2）设置 PG/PC 接口。先选定 CP 5611（PPI），再单击“确定”按钮，如图 10-26 所示。

3）更改访问路径。单击“确定”按钮，更改访问路径，如图 10-27 所示。



图 10-27　更改访问路径

4）建立 STEP 7 - Micro/WIN 与 PLC 的通信。双击“双击刷新”，如图 10-28 所示，STEP 7 - Micro/WIN 自动建立与 PLC 的通信连接，当两者通信建立成功时，弹出 PLC 的型号版本、地址和波特率信息，如图 10-29 的“1”处，再单击“确认”按钮即可。

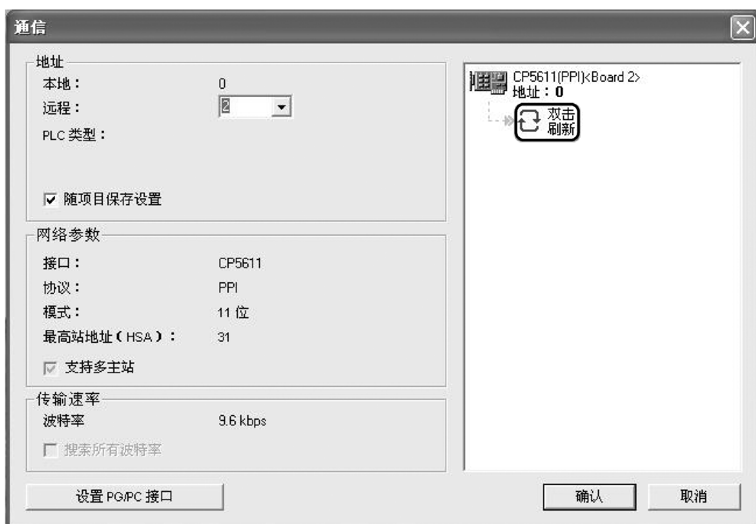


图 10-28　建立通信（1）

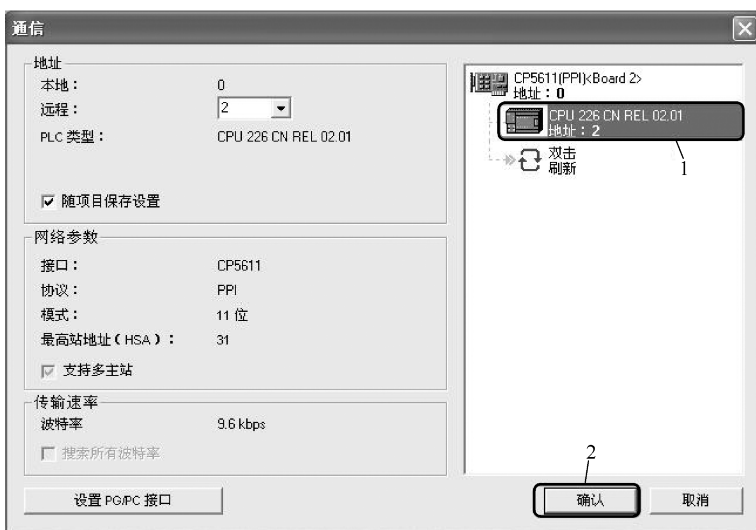


图 10-29　建立通信（2）

下载程序的方法与前述的章节相同，在此不再赘述。

【关键点】使用 CP 5611 卡下载程序，必须在计算机中安装 CP 5611 卡，CP 5611 卡是即插即用的卡，可以提供 PPI、MPI 和 PROFIBUS 通信协议与下位机进行通信。

10.2.2 用 MPI 协议下载程序

采用 MPI 通信协议下载程序多用于向 S7－300 系列 PLC 下载程序。由于 S7－200 系列 PLC 没有 MPI 接口，因此要用 MPI 协议向 S7－200 下载程序，S7－200 必须配置 EM 277 模块。可以采用两种方法，一是用 PC/MPI 适配器下载程序，另一种方法是用板卡下载程序（对于台式电脑用 CP 5611、CP 5613 卡，对于笔记本电脑用 CP 5511 卡）。

1. 用 PC/MPI 适配器下载程序

1）打开 PG/PC 接口。在界面的左侧的组件“工具”中选中“设置 PG/PC 接口”，并单击之，如图 10-16 所示。也可以在菜单中打开设置 PG/PC 接口界面，其路径为“查看”→“组件”→“设置 PG/PC 接口”。

2）设置 PG/PC 接口。先选定“PC Adapter（MPI）”，再单击“属性”按钮，如图 10-30所示。由于 PC/MPI 适配器与计算机相连的接口是 USB，因此在“本地连接”选项卡中选择“USB”，如图 10-31 所示，最后单击“确定”按钮。



图 10-30 设置 PG/PC 接口（1）



图 10-31 设置 PG/PC 接口（2）

3）更改访问路径。单击“确定”按钮，更改访问路径，如图 10-32 所示。



图 10-32 更改访问路径

4）建立 STEP 7 - Micro/WIN 与 PLC 的通信。双击“双击刷新”，如图 10-33 所示，STEP 7 - Micro/WIN 自动建立与 PLC 的通信连接，当两者通信建立成功时，弹出 PLC 的型号版本、地址信息，如图 10-34 的“1”处，再将远程中的地址选择成与找到 PLC 地址一致，本例为“3”，最后单击“确认”按钮即可。



图 10-33　建立通信（1）

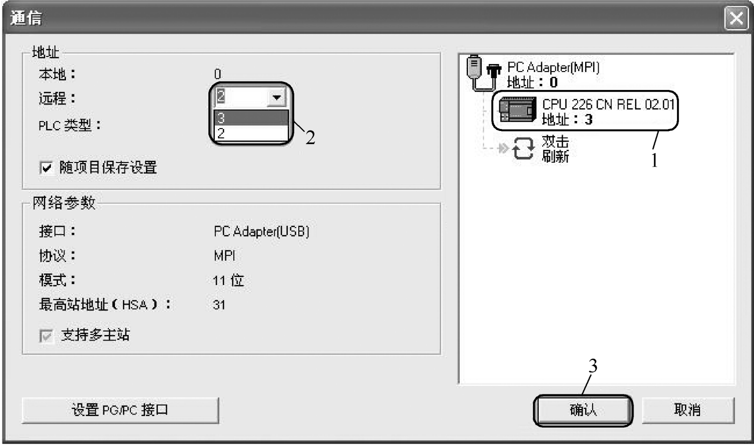


图 10-34　建立通信（2）

【关键点】图 10-34 中远程地址选择要与找到的 PLC 地址一致，否则后续的下载程序是不可能成功的。

下载程序的方法与前述的章节相同，在此不再赘述。

2. 用 CP 5611 卡下载程序

1）打开 PG/PC 接口。在界面的左侧的组件“工具”中选中“设置 PG/PC 接口”，并单击之，如图 10-16 所示。也可以在菜单中打开设置 PG/PC 接口界面，其路径为“查看”→“组件”→“设置 PG/PC 接口”。

2）设置 PG/PC 接口。先选定“CP 5611（MPI）”，再单击“确定”按钮，如图 10-35 所示。

3）更改访问路径。单击“确定”按钮，更改访问路径，如图 10-36 所示。



图 10-35 设置 PG/PC 接口



图 10-36 更改访问路径

4）建立 STEP 7 - Micro/WIN 与 PLC 的通信。双击“双击刷新”，如图 10-37 所示，STEP 7 - Micro/WIN 自动建立与 PLC 的通信连接，当两者通信建立成功时，弹出 PLC 的型号版本、地址信息，如图 10-38 的“1”处，再将远程中的地址选择成与找到 PLC 地址一致，本例为“3”，最后单击“确认”按钮即可。

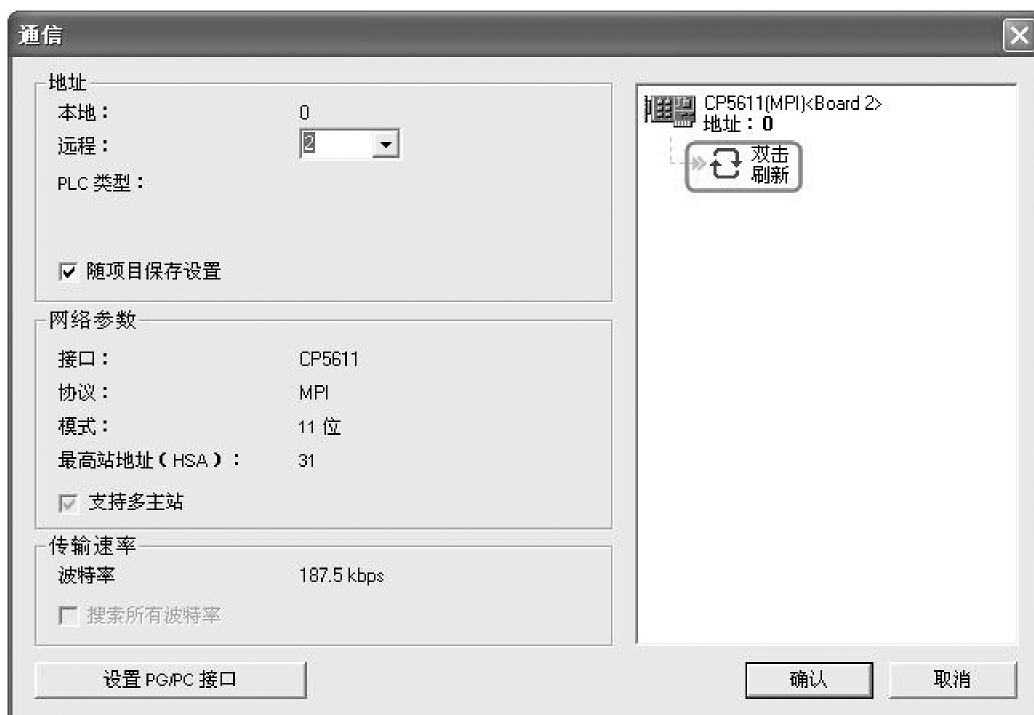


图 10-37 建立通信（1）

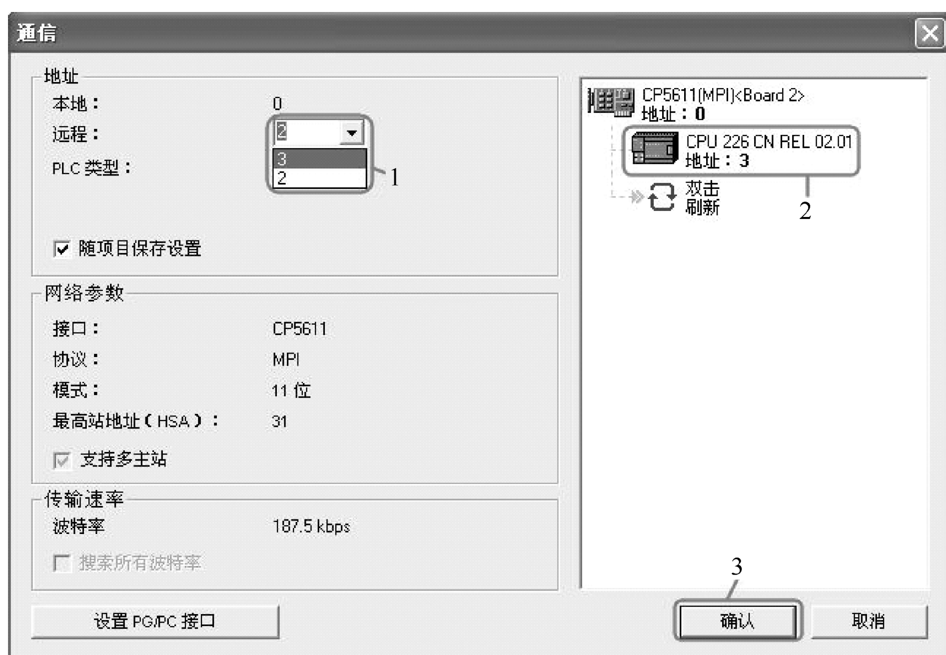


图 10-38　建立通信（2）

下载程序的方法与前述的章节相同，在此不再赘述。

顺便指出，如果控制系统中已经配置 EM 277 模块时，采用 MPI 电缆下载程序是合适的选择。如果控制系统中没有配置 EM 277 模块时，利用 MPI 适配器和 EM 277 模块向 S7 - 200 下载程序是很不经济的，因为 MPI 电缆和 EM 277 模块的价格远高于 PPI 编程电缆。

10.2.3　用 PROFIBUS 协议下载程序

采用 PROFIBUS 通信协议下载程序多用于向 S7 - 300 系列 PLC 下载程序。由于 S7 - 200 系列 PLC 没有 PROFIBUS 接口，因此要用 PROFIBUS 协议向 S7 - 200 下载程序，必须配置 EM 277 模块。计算机必须配用板卡（对于台式电脑用 CP 5611、CP 5613 卡，对于笔记本电脑用 CP 5511 卡）。下载程序步骤如下：

1）打开 PG/PC 接口。在图面的左侧的组件“工具”中选中“设置 PG/PC 接口”，并单击之，如图 10-16 所示。也可以在菜单中打开设置 PG/PC 接口界面，其路径为“查看→组件→设置 PG/PC 接口”。

2）设置 PG/PC 接口。先选定“CP 5611（PROFIBUS）”，再单击“确定”按钮，如图 10-39 所示。

3）更改访问路径。单击“确定”按钮，更改访问路径，如图 10-40 所示。

4）建立 STEP 7 - Micro/WIN 与 PLC 的通信。双击“双击刷新”，如图 10-41 所示，STEP 7 - Micro/WIN 自动建立与 PLC 的通信连接，当两者通信建立成功时，弹出 PLC 的型号版本、地址信息，如图 10-42 的“1”处，再将远程中的地址选择成与找到 PLC 地址一致，本例为“3”，最后单击“确认”按钮即可。下载程序的方法与前述的章节相同，在此不再赘述。

顺便指出，如果控制系统中已经配置 EM 277 模块时，采用 PROFIBUS 协议下载程序是合适的选择。如果控制系统中没有配置 EM 277 模块时，利用 PROFIBUS 协议和 EM 277 模块向 S7－200 下载程序是很不经济的。

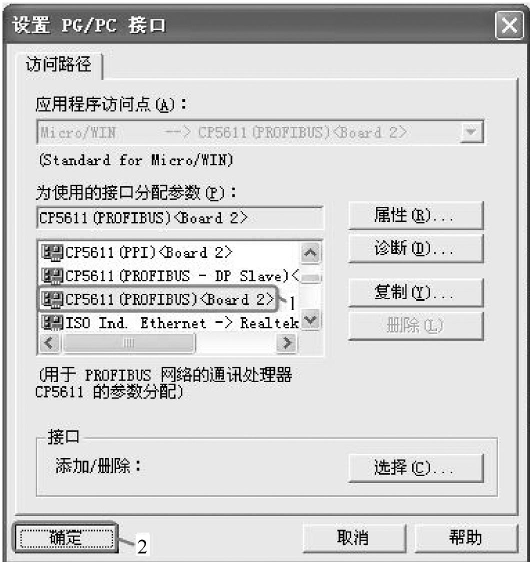


图 10-39　设置 PG/PC 接口



图 10-40　更改访问路径



图 10-41　建立通信（1）

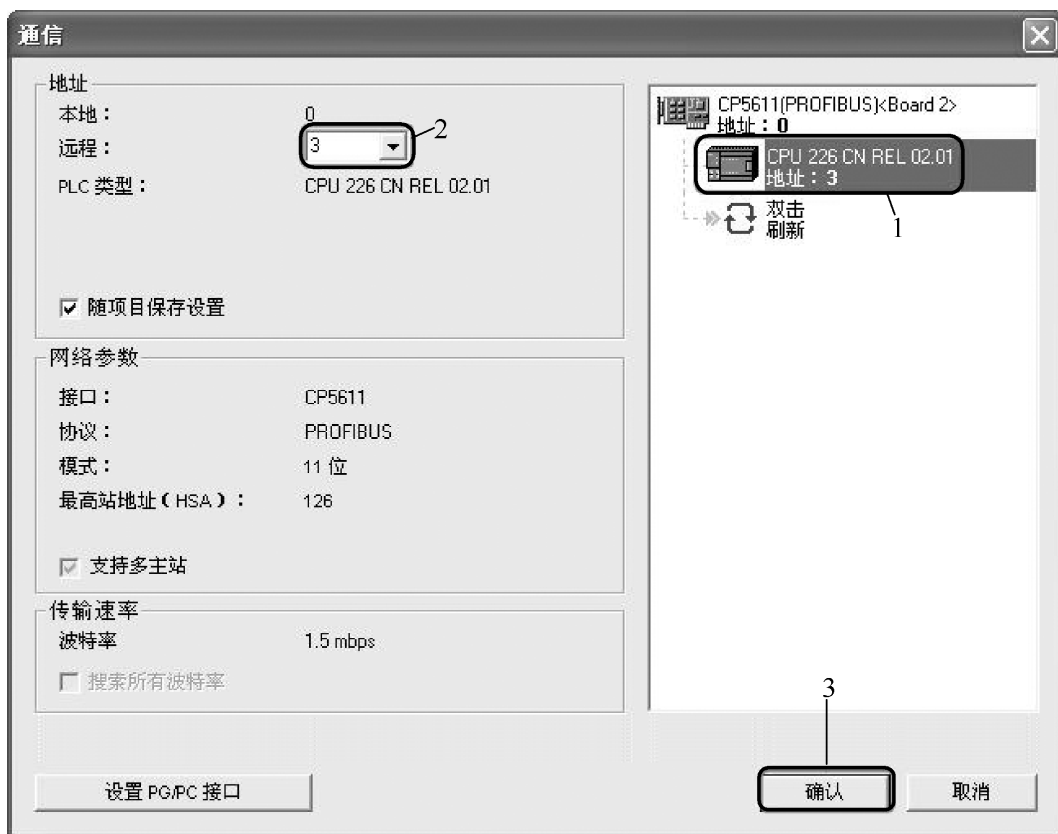


图 10-42　建立通信（2）

10.2.4　用 TCP/IP 下载程序

由于 S7-200 系列 PLC 没有以太网接口，因此要用 TCP/IP 向 S7-200 下载程序，必须配置以太网模块（如 CP 243-1 IT 模块）。需要指出的是 STEP 7-Micro/WIN V4.0 与 PLC 的 TCP/IP 通信的前提是 PLC 与以太网模块的硬件配置已经通过其他方式（如 PPI 通信）下载到 PLC 中。以下将介绍用 TCP/IP 向 S7-200 下载程序的方法。

1）首先建立 STEP 7-Micro/WIN V4.0 与 S7-200 之间的通信。具体的方法与“用 PC/PPI 编程电缆下载程序”的前 6 步完全相同。

2）打开“以太网向导”。单击“工具”下的子菜单“以太网向导”，弹出“以太网向导”，如图 10-43、图 10-44 所示，单击“下一步”按钮。

3）指定模块位置。在模块位置中输入位置号，本例为“0”，再单击“读取模块”按钮，若读取成功，则模块的信息显示在如图 10-45 的序号“3”处。

4）指定模块地址。在“IP 地址”中输入“192.168.0.2”（也可以是其他有效 IP 地址），在“子网掩码”中输入“255.255.255.0”，“网关”地址可以空置，“为此模块指定通信连接类型”中选择“自动检测通信”选项，最后单击“下一步”按钮，如图 10-46 所示。



图 10-43　打开以太网向导

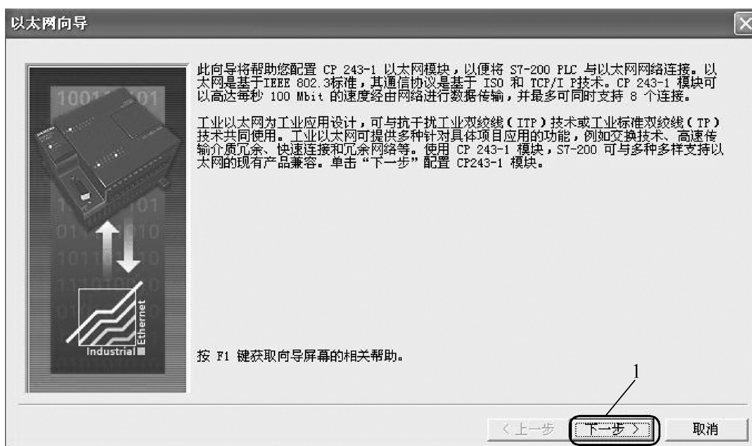


图 10-44　以太网向导界面

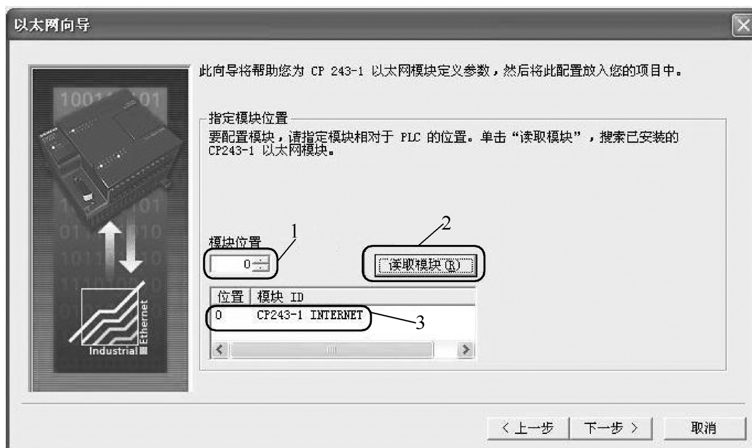


图 10-45　指定模块位置

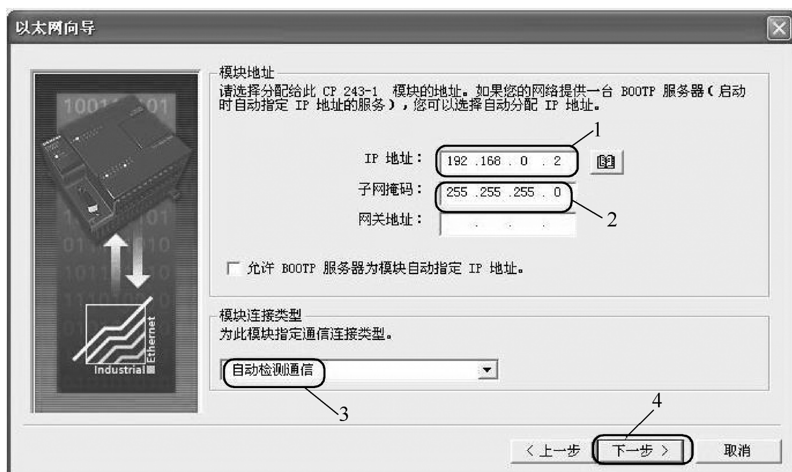


图 10-46　指定模块地址

5）指定连接数目。因为只有两个模块，所以“为此模块配置的连接数”选定为“1”，再单击“下一步”按钮，如图 10-47 所示。

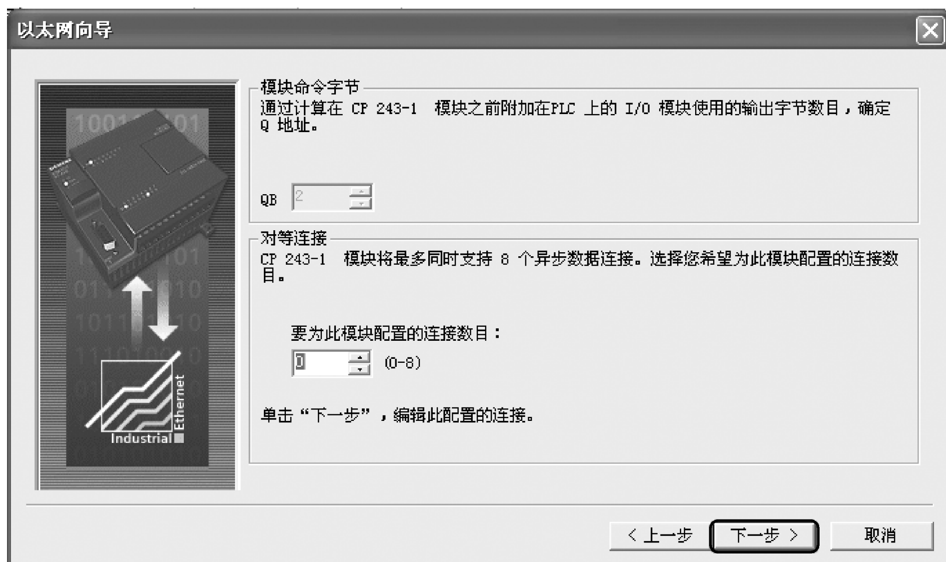


图 10-47　指定连接数目

6）CRC 保护与保持现用间隔。选中序号“1”处的选项，单击“下一步”按钮，如图 10-48 所示。

7）分配配置内存。单击“建议地址”按钮，再单击“下一步”按钮，如图 10-49 所示。注意生成的建议地址 VB159 ~ VB317 供通信使用，读者编程不可使用。

8）生成项目部件。单击“完成”按钮，以太网向导完成，STEP 7 - Micro/WIN 自动生成子程序，如图 10-50所示。

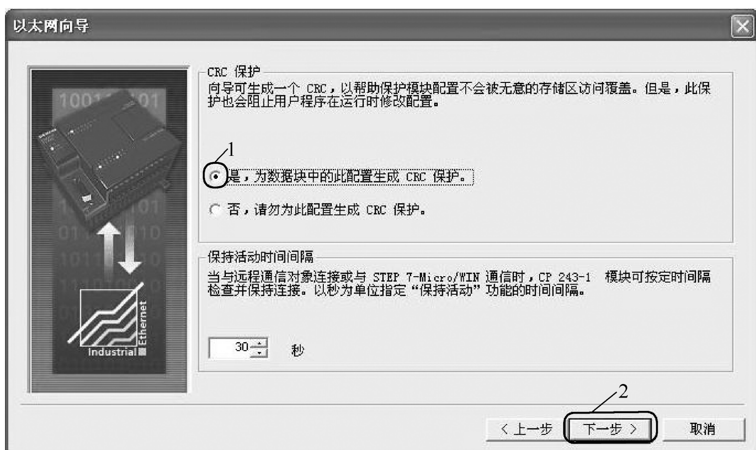


图 10-48 CRC 保护与保持现用间隔

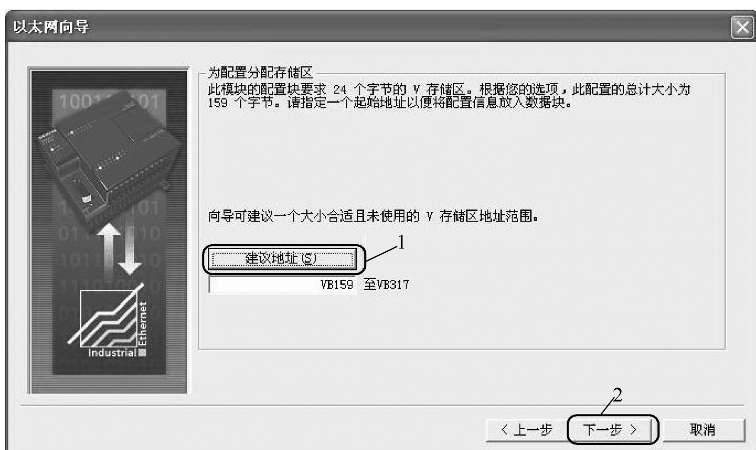


图 10-49 分配配置内存

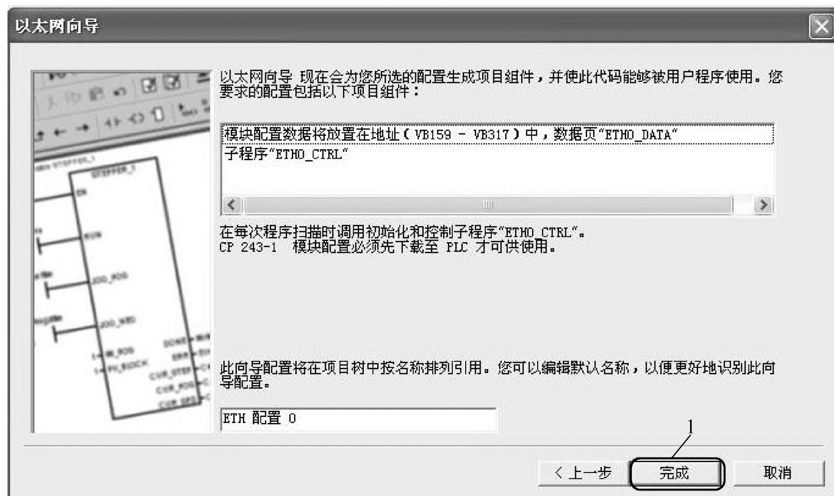


图 10-50 生成项目部件

9）将配置下载到 PLC。下载的方法与“用 PC/PPI 编程电缆下载程序”的第 7 步相同。

10）设置 PG/PC 接口。先单击“设置 PG/PC 接口”，再选定“TCP/IP”选项，最后单击“属性”按钮，如图 10-51所示。

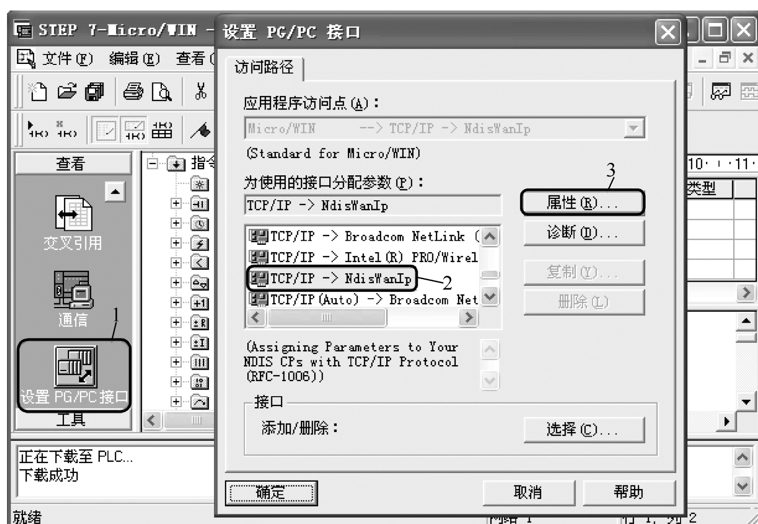


图 10-51　设置 PG/PC 接口

11）设置 PC 的 IP 地址。单击“网络属性”按钮，如图 10-52 所示。弹出网络连接如图 10-53 所示，双击“本地连接”，弹出设定 PC 的 IP 地址界面如图 10-54 所示，设置计算机的 IP 地址为“192. 168. 0. 8”，子网掩码为“255. 255. 255. 0”，单击“确定”按钮。回到如图 10-52 所示的界面，单击“确定”按钮。

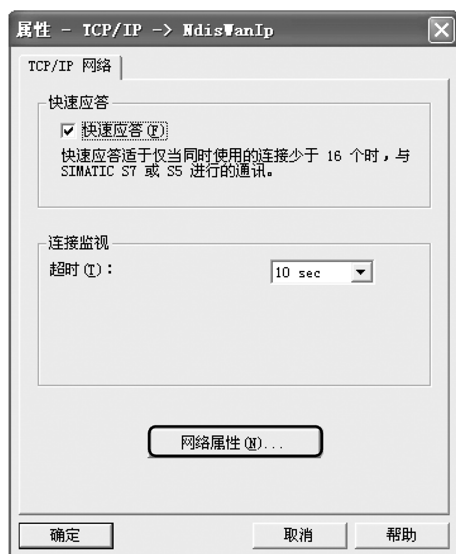


图 10-52　TCP/IP 属性



图 10-53　网络连接

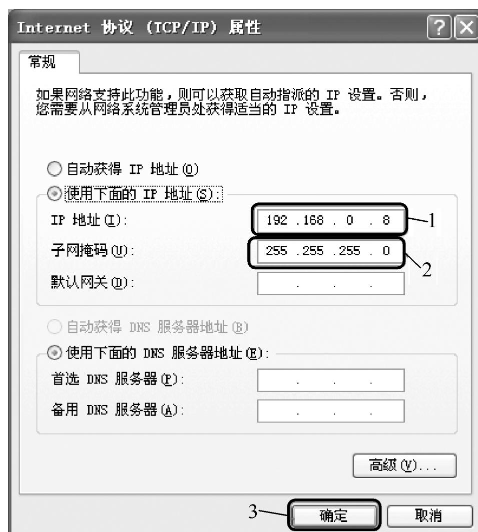


图 10-54　设定 PC 的 IP 地址

【关键点】计算机的 IP 地址必须与 CP 243－1 的 IP 地址在同一网段，即末位不同其余位相同，子网掩码则应完全应相同。

12）建立连接。在“远程”中设定 CP 243－1 IT 的 IP 地址为“192.168.0.2”，双击“双击刷新”，如图 10-55 所示。当 STEP 7－Micro/WIN V4.0 与 CP 243－1 已经建立连接时，弹出 PLC 的型号版本和 IP 地址，如图 10-56 所示，单击“确认”按钮即可。

13）下载程序。单击“下载”按钮，如图 10-57 所示，弹出如图 10-58 所示下载界面，单击“下载”按钮程序开始下载。正在下载程序的状态，如图 10-59 所示。

顺便指出，如果为下载程序而配置 CP 243－1 模块是不必要的，但如果控制系统中已经配置 CP 243－1 模块时，采用以太网下载程序是合适的选择。用以太网下载程序的速度明显比 PPI 电缆下载程序快得多。



图 10–55　建立连接（1）

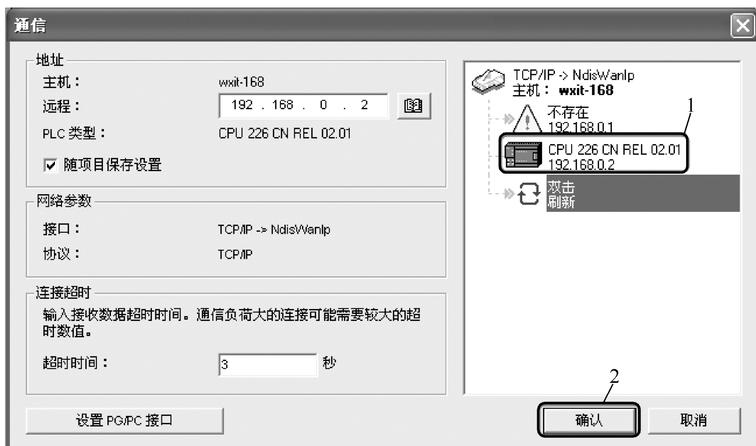


图 10–56　建立连接（2）

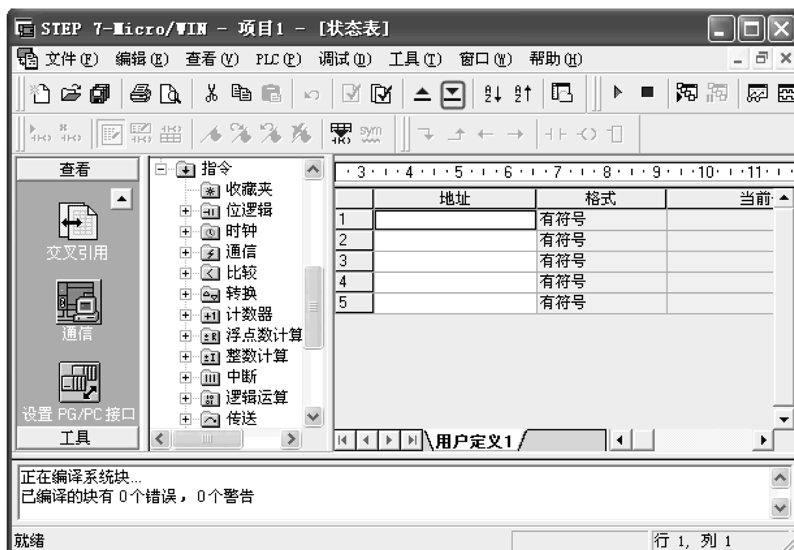


图 10–57　下载（1）

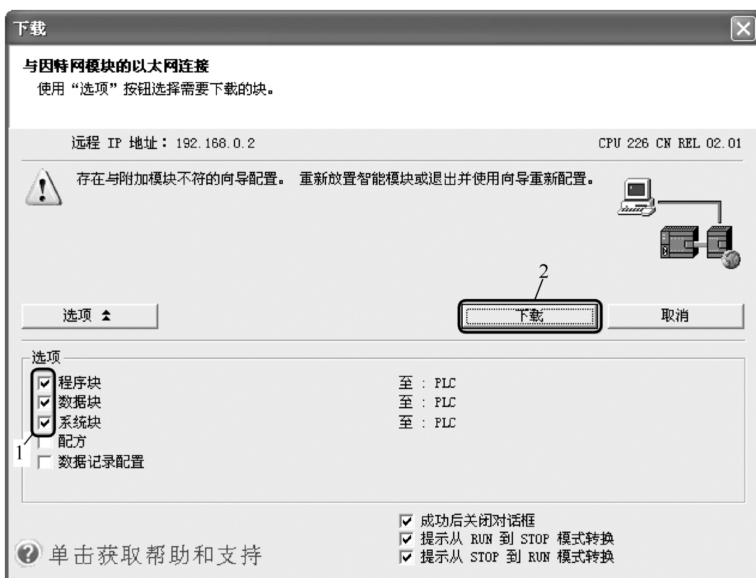


图 10-58　下载（2）



图 10-59　正在下载

【关键点】 用以太网接口（或者以太网模块）下载程序是比较常见的，而且成为一种趋势，在大中型 PLC 中（如 S7－300/400，RX3i 等）更是常用，不同的 PLC 用以太网下载程

序的步骤基本类似，因此读者应该掌握这种方法。

小　　结

（1）对 PID 概念的理解和 PID 三个参数的调节。
（2）PID 指令的各参数的含义。
（3）程序的各种下载方法。

习　　题

1. PID 三个参数的含义是什么？
2. 闭环控制有什么特点？
3. 简述调整 PID 三个参数的方法。
4. 简述 PID 控制器的主要优点。
5. 某水箱的出水口的流量是变化的，注水口的流量可通过调节水泵的转速控制，水位的检测可以通过水位传感器完成，水箱最大盛水高度为 2 m，要求对水箱进行水位控制，保证水位高度为 1.6 m。用 PLC 作为控制器，EM 231 为模拟量输入模块，用于测量水位信号，用 EM 232 产生输出信号，控制变频器，从而控制水泵的输出流量。水箱的水位控制的原理图如图 10-60 所示。

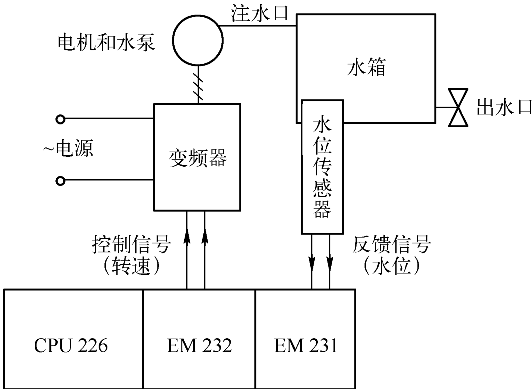


图 10-60　水箱的水位控制的原理图

第 11 章　工程应用案例

本章介绍 11 个典型的可编程序控制器系统集成的案例，且都是实际的工程案例，供读者模仿学习，本章是前面章节内容的综合应用，因此本章的例题都有一定的难度。

11.1　分高峰和非高峰时段的交通灯 PLC 控制系统

【例 11-1】某十字路口的交通灯，如图 11-1 所示。其中，R、Y、G 分别代表红、黄、绿的交通灯。要完成如下功能：

1）设置起动按钮、停止按钮。正常起动情况下，东西向绿灯亮 30 s，转东西向绿灯以 0.5 s 闪烁 4 s，转东西向黄灯亮 3 s，转南北向绿灯亮 30 s，转南北向绿灯以 0.5 s 闪烁 4 s，转南北向黄灯亮 3 s，再转东西向绿灯亮30 s，以此类推。

2）假设 PLC 内部时钟为北京时间，在上午7：30 ~9：00 及下午 16：30 ~18：00 为上班高峰时段，在这一时间段内，绿灯的常亮时间为 45 s，其余闪烁及黄灯时间不变。

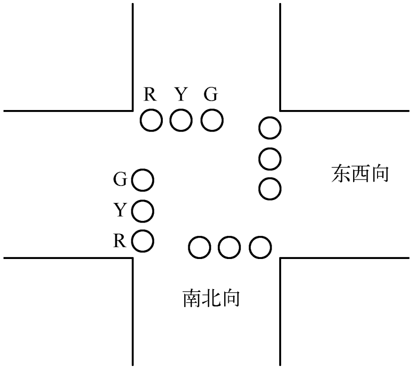


图 11-1　交通灯示意图

3）在东西向绿灯亮时，南北向应显示红灯。同理，南北向绿灯亮时，东西向应显示红灯。

11.1.1　绘制时序图

由于十字交通灯的逻辑比较复杂，为了方便编写程序，可先根据题意绘制时序图，如图 11-2 所示。

把不同颜色的灯的亮灭情况，罗列出来，具体如下：

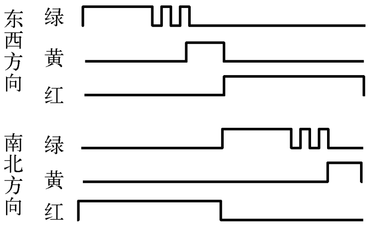


图 11-2　交通灯时序图

1. 正常时段

（1）东西方向

$T<30$ s，绿灯亮；30 s≤$T<34$ s，绿灯闪烁；34 s≤$T<37$ s，黄灯亮；37 s≥T≥74 s，红灯亮。

（2）南北方向

$T<37$ s，红灯亮；37 s≤$T<67$ s，绿灯亮；67 s≤$T<71$ s，绿灯闪烁；74 s≥T≥71 s，黄灯亮。

2. 高峰时段

（1）东西方向

$T<45$ s，绿灯亮；45 s≤$T<49$ s，绿灯闪烁；49 s≤$T<52$ s，黄灯亮；104 s≥T≥52 s，

红灯亮。

（2）南北方向

$T<52$ s，红灯亮；52 s$\leqslant T<97$ s，绿灯亮；97 s$\leqslant T<101$ s，绿灯闪烁；104 s$\geqslant T\geqslant$101 s，黄灯亮。

11.1.2　系统软硬件配置

1. 系统的软硬件

① 1 台 CPU 226CN。

② 1 套 STEP 7 - Micro/Win V4.0 SP9。

③ 1 根 PC/PPI 电缆。

2. 控制系统的接线

交通灯控制系统的接线比较简单，如图 11-3 所示。

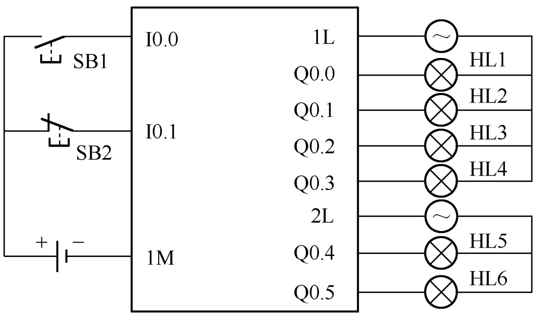


图 11-3　PLC 接线图

3. PLC 的 I/O 分配

PLC 的 I/O 分配见表 11-1。

表 11-1　PLC 的 I/O 分配表

输　入			输　出		
名 称	符 号	输入点	名 称	符 号	输出点
开始按钮	SB1	I0.0	绿灯（东西）	HL1	Q0.0
停止按钮	SB2	I0.1	黄灯（东西）	HL2	Q0.1
			红灯（东西）	HL3	Q0.2
			绿灯（南北）	HL4	Q0.3
			黄灯（南北）	HL5	Q0.4
			红灯（南北）	HL6	Q0.5

11.1.3　编写控制程序

编写这个程序容易想到 2 种方法，一是比较指令编写，梯形图如图 11-4 所示，相对简

单，二是基本指令编写，梯形图如图 11-5 所示，相对麻烦。

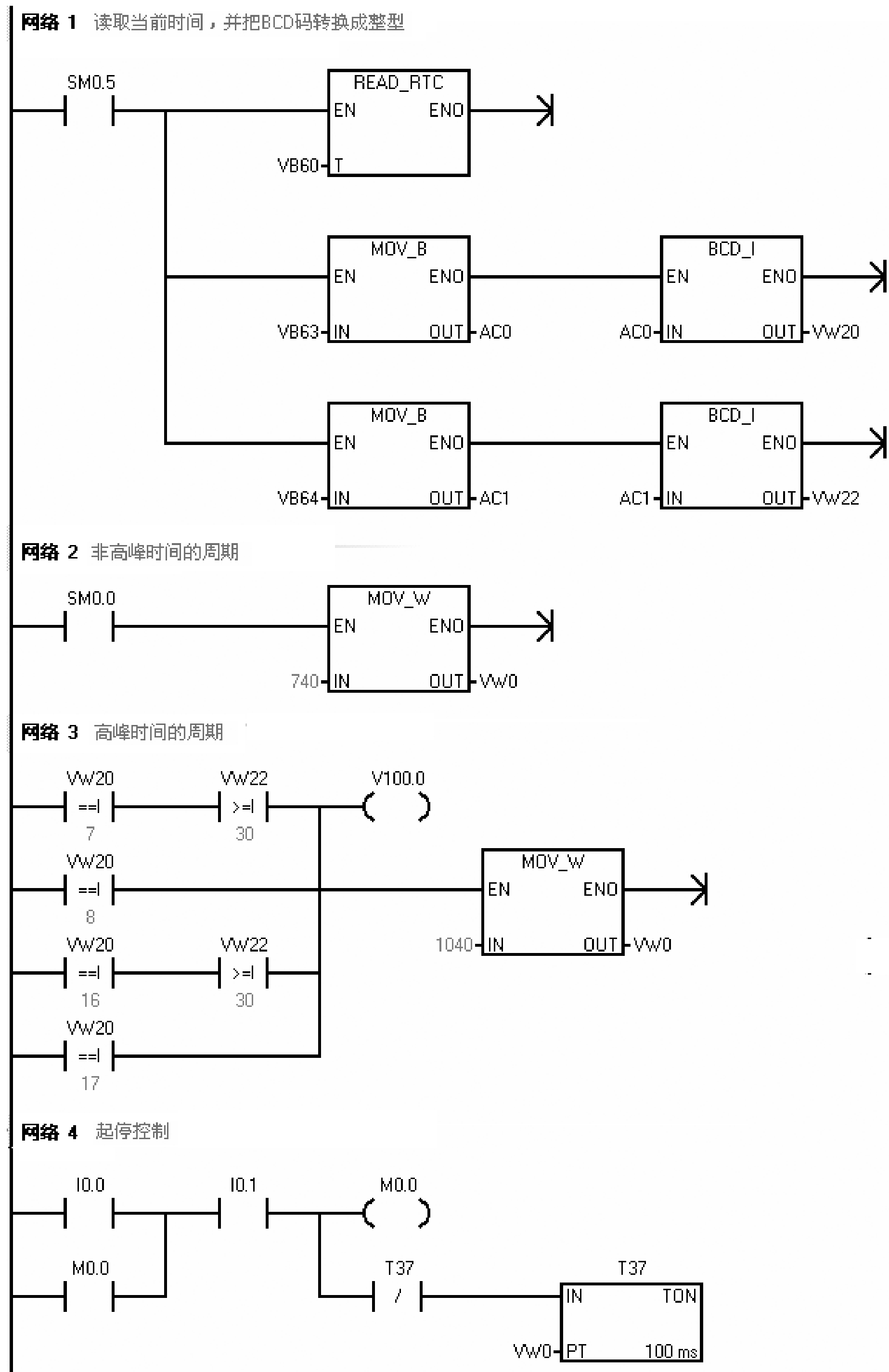
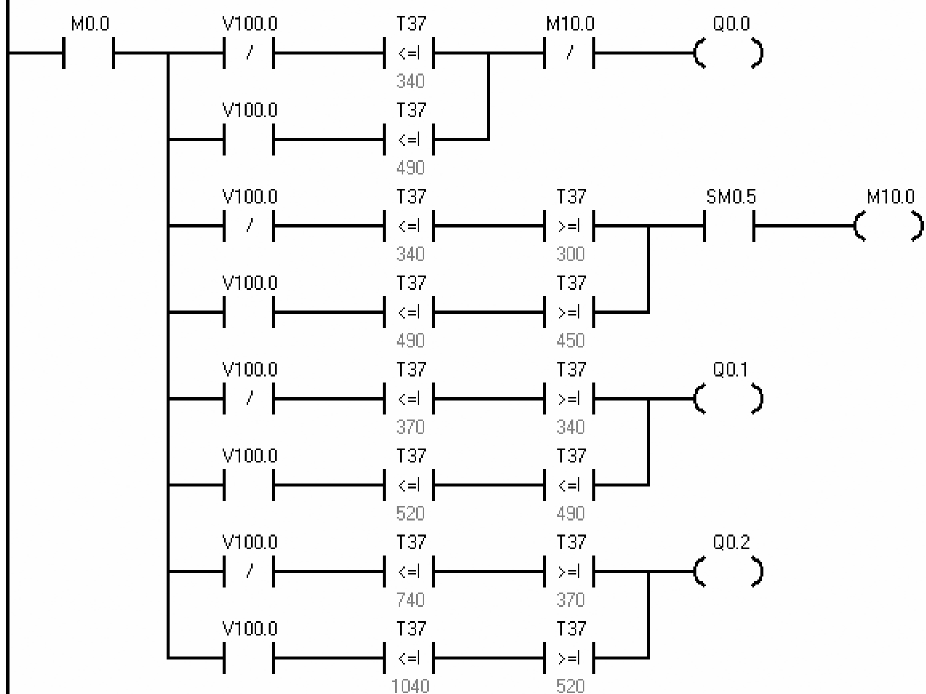


图 11-4 梯形图（比较指令）



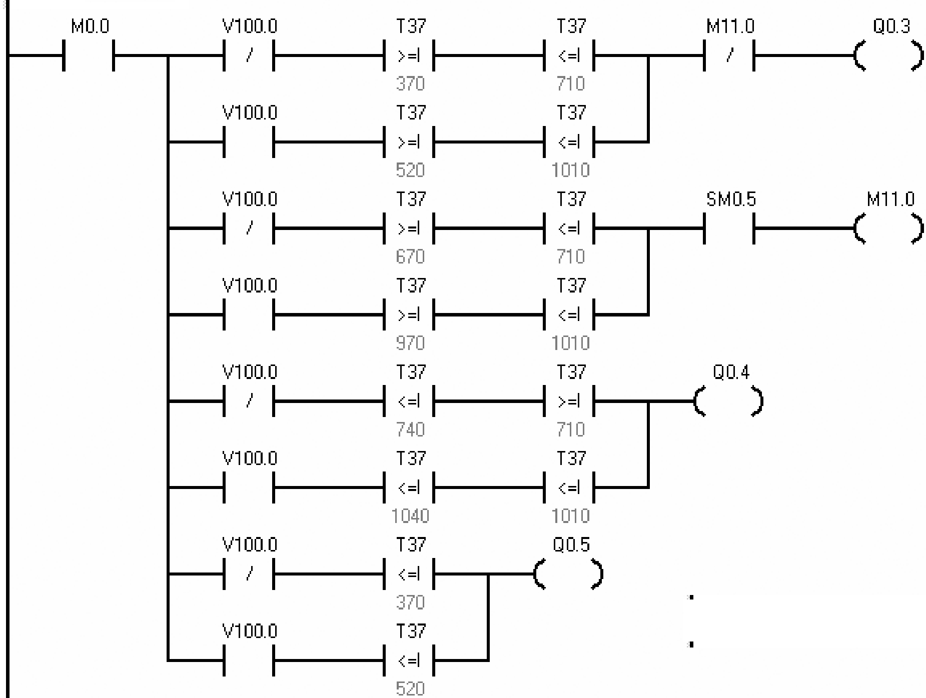


图 11-4 梯形图（比较指令）（续）

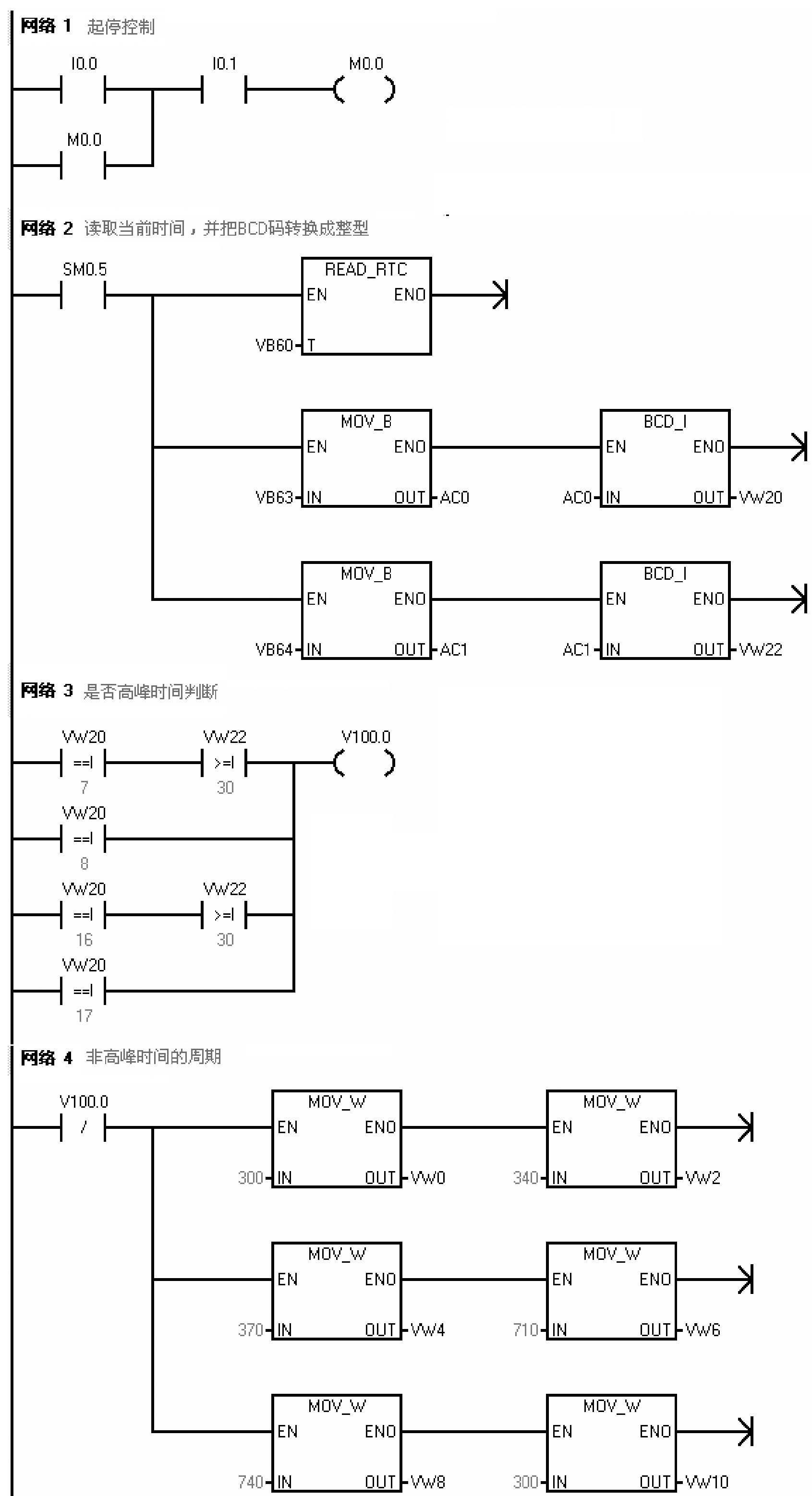


图 11-5　梯形图（基本指令）

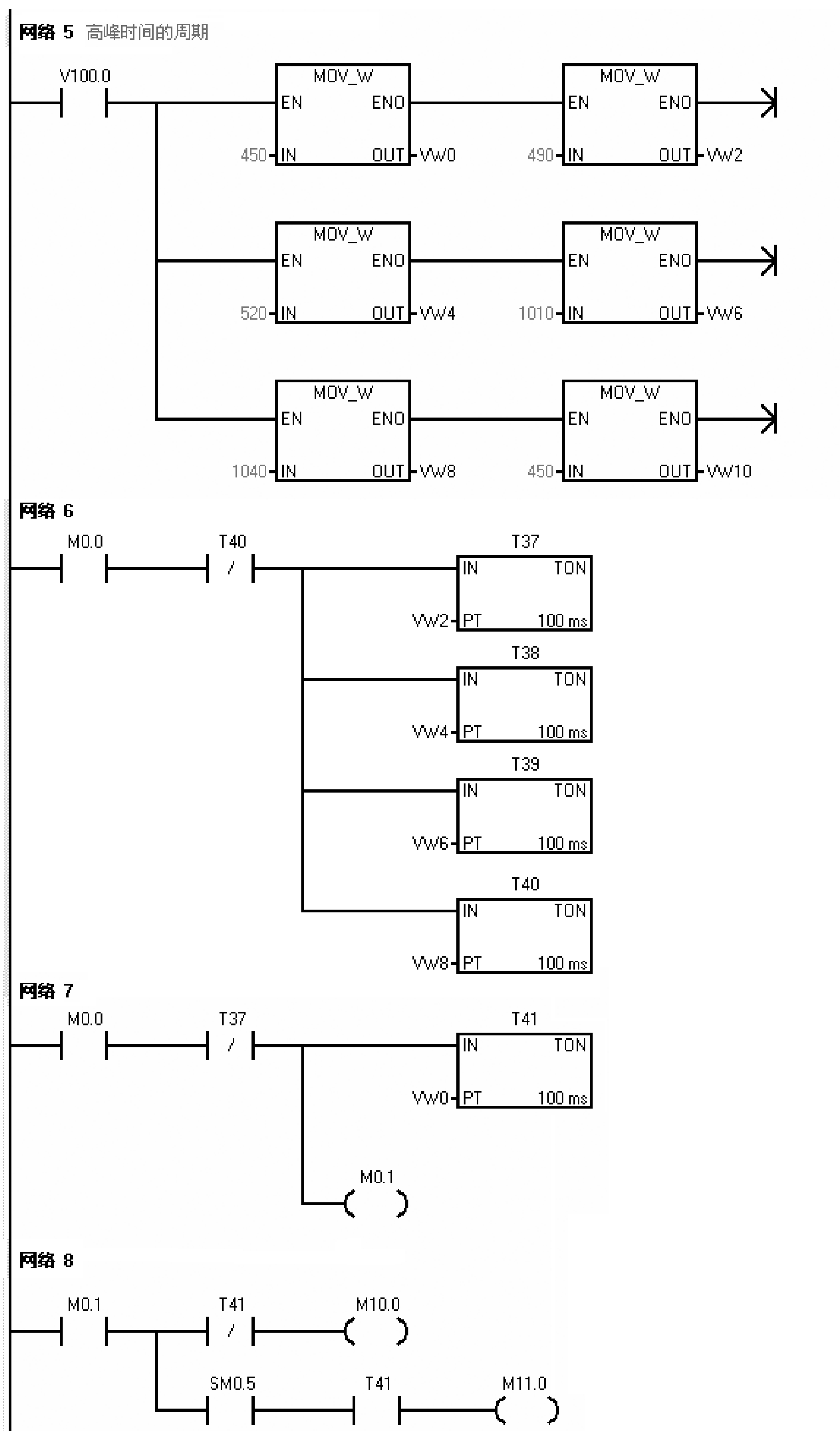


图 11-5 梯形图（基本指令）（续）

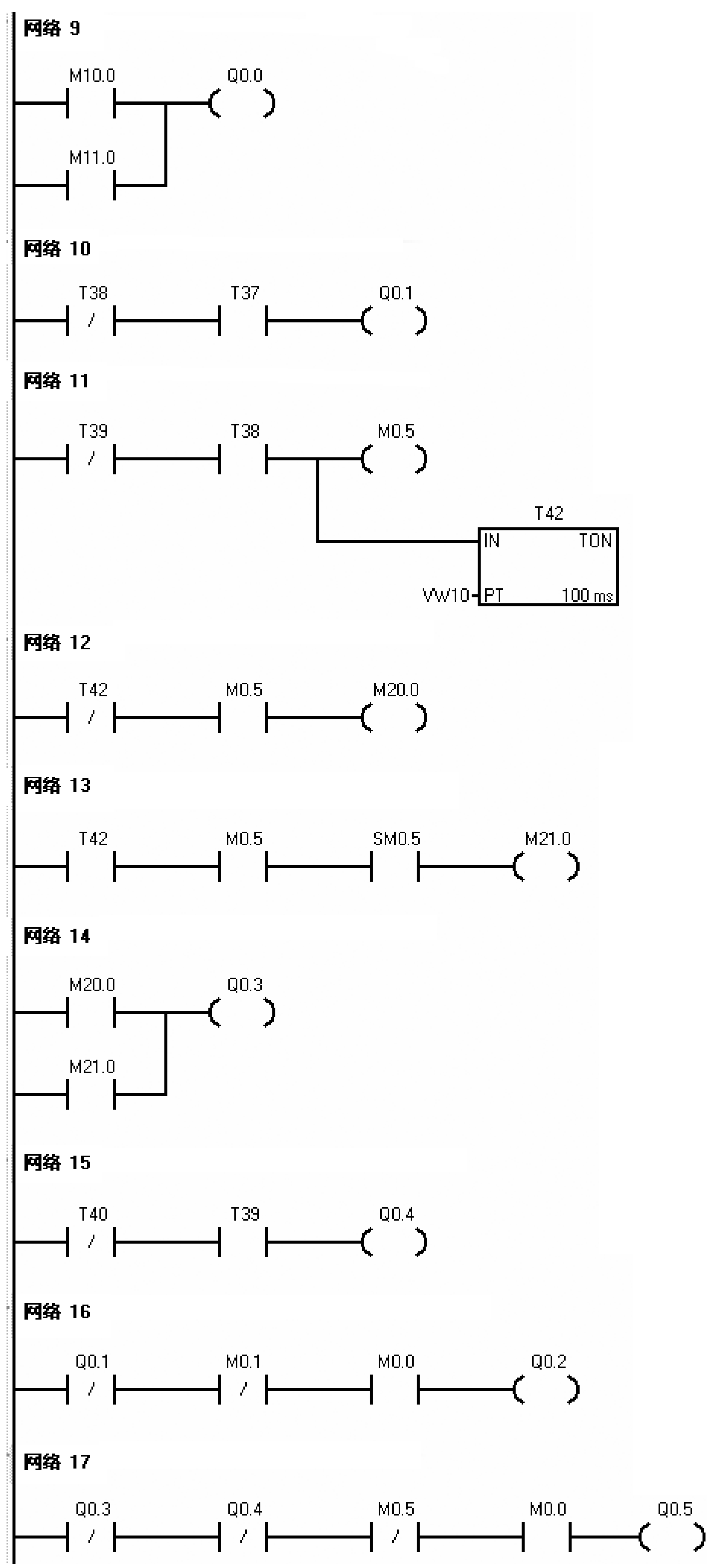


图 11-5　梯形图（基本指令）（续）

11.2 行车呼叫 PLC 控制系统

【例 11-2】如图 11-6 所示，为行车呼车系统示意图。一部电动运输车提供 8 个工位使用。系统共有 12 个按钮。图中，SB1 ~ SB8 为每一工位的呼车按钮。SB9、SB10 为电动小车点动左行，点动右行按钮。SB11、SB12 为起动和停止按钮。系统上电后，可以按下这两个按钮调整小车位置，使小车停于工位位置。SQ1 ~ SQ8 为每一工位信号。

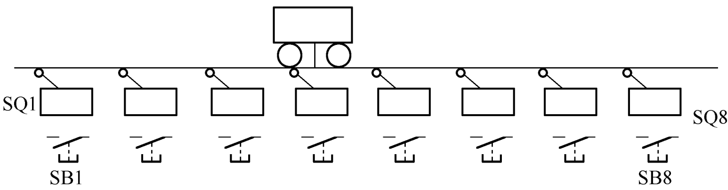


图 11-6 行车呼叫示意图

正常工作流程如下：小车在某一工位，若无呼车信号，除本工位指示灯不亮外，其余指示灯亮，表示允许呼车。当某工位呼车按钮按下，各工位指示灯全部熄灭，行车运动至该车位，运动期间呼车按钮失效。呼车工位号大于停车位时，小车右行，反之则左行。当小车停在某一工位后，停车时间为 30 s，以便处理该工位工作流程。在此段时间内，其他呼车信号无效。从安全角度考虑，停电再来电后，小车不允许运行。

试：

① 列出该系统的 I/O 配置表。

② 编写 PLC 程序并进行调试。

11.2.1 系统软硬件配置

1. 系统的软硬件

① 1 台 CPU 226CN。

② 1 套 STEP 7 - Micro/Win V4.0 SP9。

③ 1 根 PC/PPI 电缆。

2. PLC 的 I/O 分配

PLC 的 I/O 分配见表 11-2。

表 11-2 PLC 的 I/O 分配表

名　称	符　号	输　入　点	名　称	符　号	输　出　点
1 号位置呼叫按钮	SB1	I0.0	电动机正转	KA1	Q0.0
2 号位置呼叫按钮	SB2	I0.1	电动机反转	KA2	Q0.1
3 号位置呼叫按钮	SB3	I0.2	指示灯	HL1	Q0.2
4 号位置呼叫按钮	SB4	I0.3			
5 号位置呼叫按钮	SB5	I0.4			
6 号位置呼叫按钮	SB6	I0.5			
7 号位置呼叫按钮	SB7	I0.6			

（续）

名　称	符　号	输 入 点	名　称	符　号	输 出 点
8 号位置呼叫按钮	SB8	I0. 7			
行车位于 1 号位置	SQ1	I1. 0			
行车位于 2 号位置	SQ2	I1. 1			
行车位于 3 号位置	SQ3	I1. 2			
行车位于 4 号位置	SQ4	I1. 3			
行车位于 5 号位置	SQ5	I1. 4			
行车位于 6 号位置	SQ6	I1. 5			
行车位于 7 号位置	SQ7	I1. 6			
行车位于 8 号位置	SQ8	I1. 7			
点动按钮（正）	SB9	I2. 0			
点动按钮（反）	SB10	I2. 1			
起动按钮	SB11	I2. 2			
停止按钮	SB12	I2. 3			

3. 控制系统的接线

控制系统的接线如图 11-7 所示。

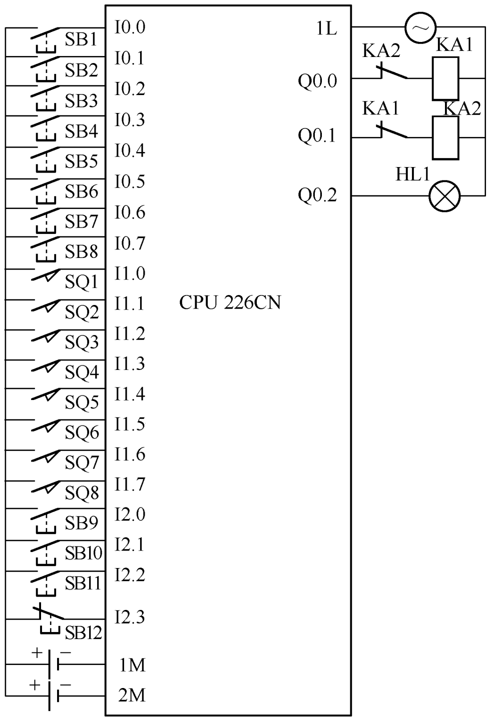


图 11-7　PLC 接线图

11. 2. 2　编写程序

系统的主程序如图 11-8 所示，子程序如图 11-9 所示。

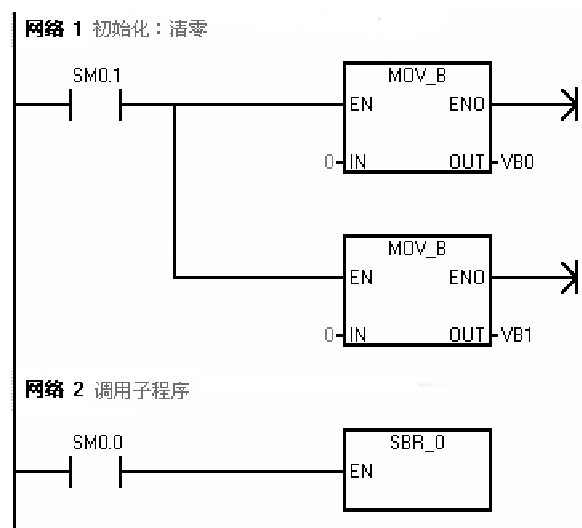


图 11-8　主程序

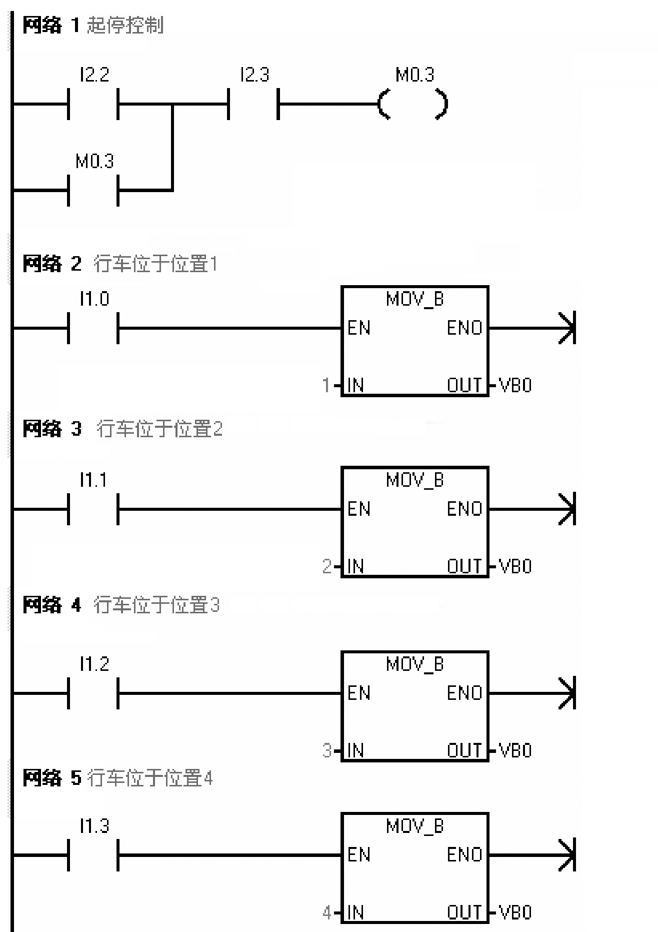
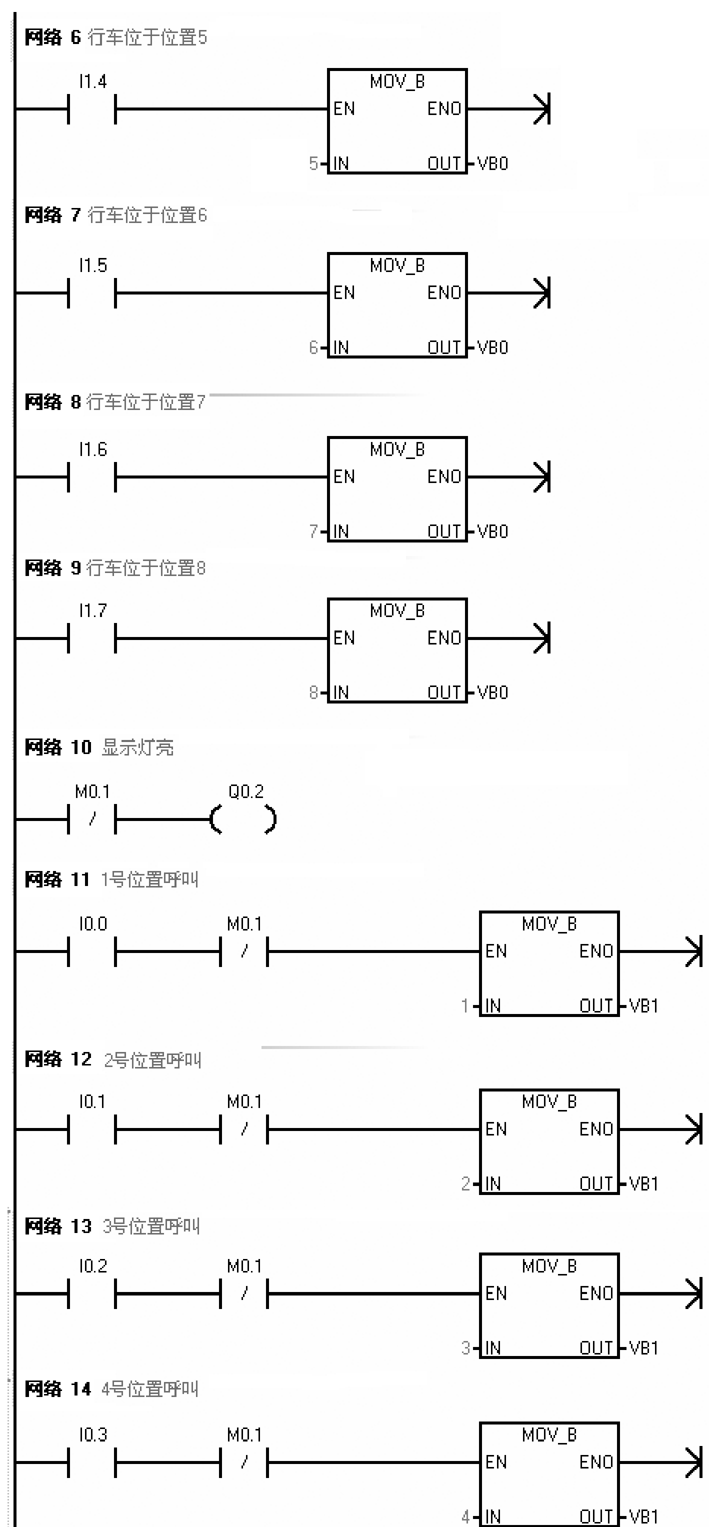


图 11-9　子程序



网络 6 行车位于位置5
I1.4
MOV_B
EN
ENO
5
IN
OUT
VB0
网络 7 行车位于位置6
I1.5
MOV_B
EN
ENO
6
IN
OUT
VB0
网络 8 行车位于位置7
I1.6
MOV_B
EN
ENO
7
IN
OUT
VB0
网络 9 行车位于位置8
I1.7
MOV_B
EN
ENO
8
IN
OUT
VB0
网络 10 显示灯亮
M0.1
Q0.2
网络 11 1号位置呼叫
I0.0
M0.1
MOV_B
EN
ENO
1
IN
OUT
VB1
网络 12 2号位置呼叫
I0.1
M0.1
MOV_B
EN
ENO
2
IN
OUT
VB1
网络 13 3号位置呼叫
I0.2
M0.1
MOV_B
EN
ENO
3
IN
OUT
VB1
网络 14 4号位置呼叫
I0.3
M0.1
MOV_B
EN
ENO
4
IN
OUT
VB1

图 11–9　子程序（续）

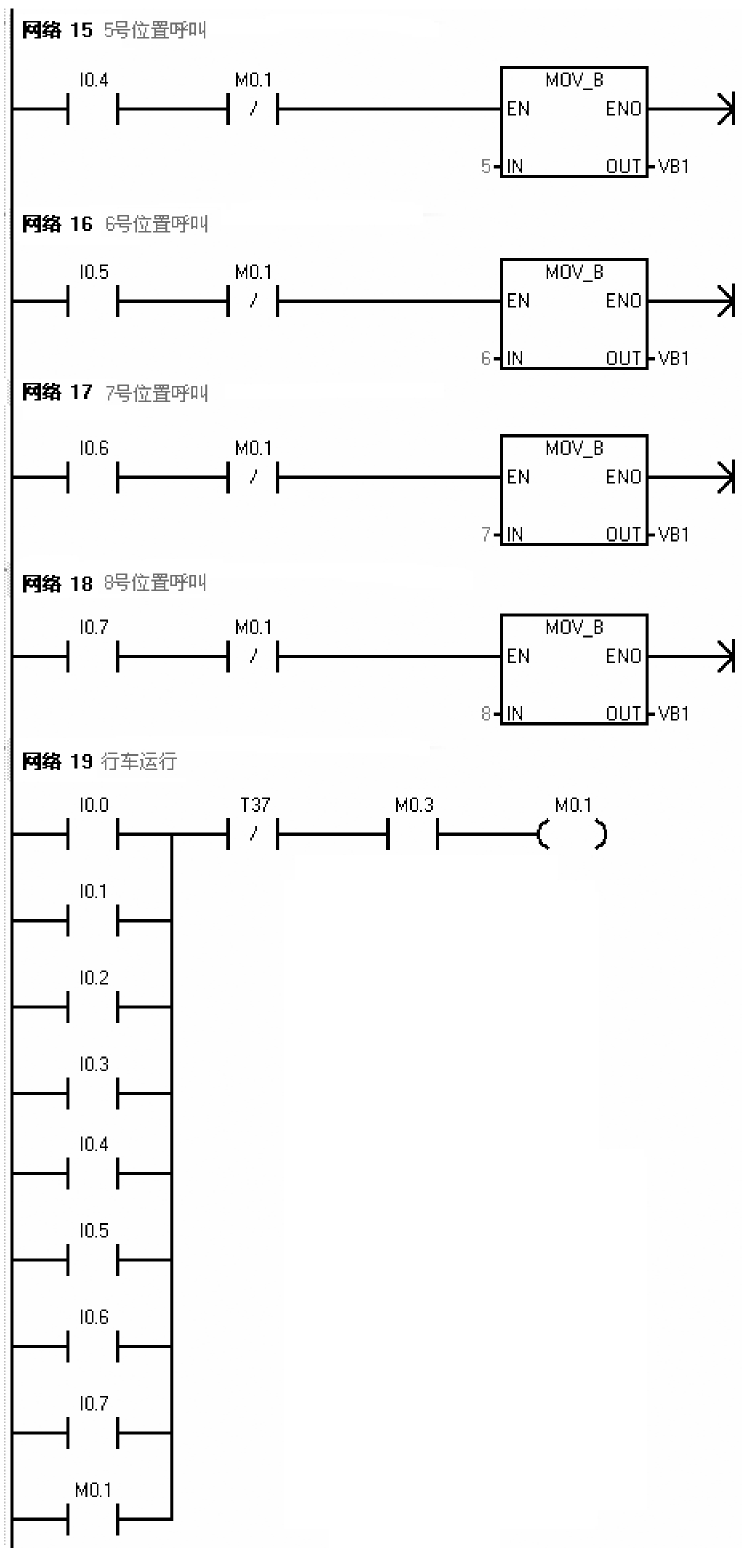

网络 15 5号位置呼叫
I0.4
M0.1
MOV_B
EN
ENO
5
IN
OUT
VB1
网络 16 6号位置呼叫
I0.5
M0.1
MOV_B
EN
ENO
6
IN
OUT
VB1
网络 17 7号位置呼叫
I0.6
M0.1
MOV_B
EN
ENO
7
IN
OUT
VB1
网络 18 8号位置呼叫
I0.7
M0.1
MOV_B
EN
ENO
8
IN
OUT
VB1
网络 19 行车运行
I0.0
T37
M0.3
M0.1
I0.1
I0.2
I0.3
I0.4
I0.5
I0.6
I0.7
M0.1

图 11-9　子程序（续）

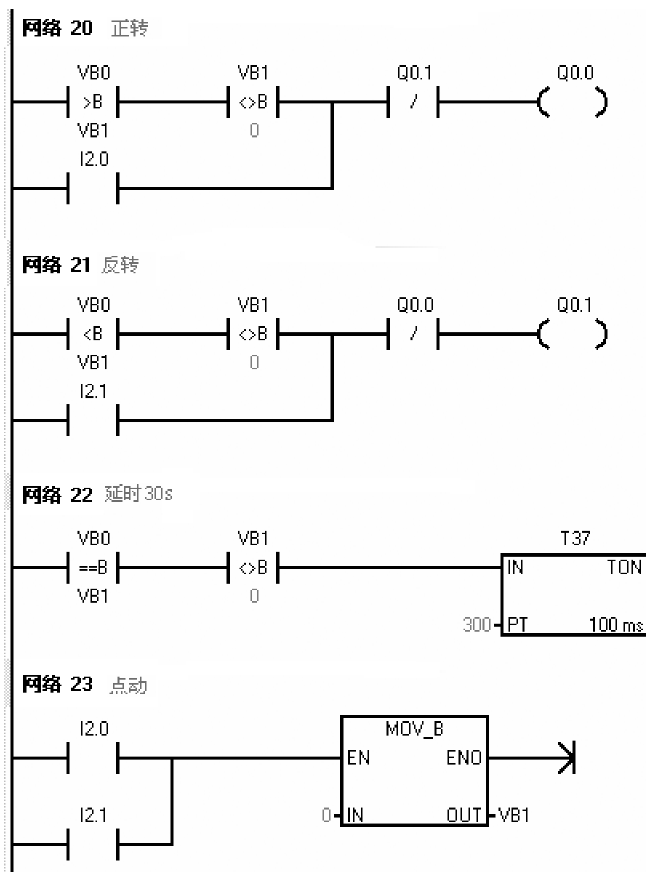


图 11-9　子程序（续）

11.3　送料小车自动往复运动的 PLC 控制系统

【例 11-3】现有一套送料小车系统，分别在工位 1、工位 2、工位 3 这三个地方来回自动送料，小车的运动由一台交流电动机进行控制。在三个工位处，分别装置了三个传感器 SQ1、SQ2、SQ3，用于检测小车的位置。在小车运行的左端和右端分别安装了两个行程开关 SQ4、SQ5，用于定位小车的原点和右极限位点。

其结构示意图如图 11-10 所示。

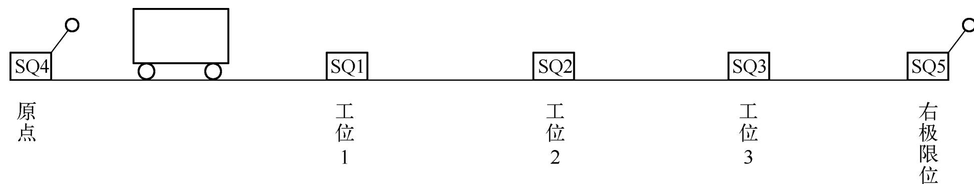


图 11-10　结构示意图

控制要求如下：

1）当系统上电时，无论小车处于何种状态，首先回到原点准备装料，等待系统的起动。

2）当系统的手/自动转换开关打开自动运行挡时，按下起动按钮 SB1，小车首先正向运行到工位 1 的位置，等待 10 s 卸料完成后正向运行到工位 2 的位置，等待 10 s 卸料完成后正向运行到工位 3 的位置，停止 5 s 后接着反向运行到工位 2 的位置，停止 5 s 后再反向运行到工位 1 的位置，停止 5 s 后再反向运行到原点位置，等待下一轮的起动运行。

3）当按下停止按钮 SB2 时系统停止运行，如果电动机停止在某一工位，则小车继续停止等待；当小车正运行在去往某一工位的途中，则当小车到达目的地后再停止运行。再次按下起动按钮 SB1 后，设备按剩下的流程继续运行。

4）当系统按下急停按钮 SB5 时，小车立即要求停止工作，直到急停按钮取消时，系统恢复到当前状态。

5）当系统的手/自动转换开关打开手动运行挡时，可以通过手动按钮 SB3、SB4 控制小车的正/反向运行。

11.3.1 系统软硬件配置

1. 系统的软硬件配置

① 1 台 CPU 226CN。

② 1 套 STEP 7 - Micro/Win V4.0 SP9。

③ 1 根 PC/PPI 电缆。

2. PLC 的 I/O 分配

PLC 的 I/O 分配见表 11-3。

表 11-3 PLC 的 I/O 分配表

名 称	符 号	输 入 点	名 称	符 号	输 出 点
起动	SB1	I0.0	电动机正转	KA1	Q0.0
停止	SB2	I0.1	电动机反转	KA2	Q0.1
左点动	SB3	I0.2			
右点动	SB4	I0.3			
工位 1	SQ1	I1.0			
工位 2	SQ2	I1.1			
工位 3	SQ3	I1.2			
原位	SQ4	I1.3			
右限位	SQ5	I1.4			
手/自	SA1	I2.0			
急停	SB5	I2.1			

3. 控制系统的接线

接线图如图 11-11 所示。

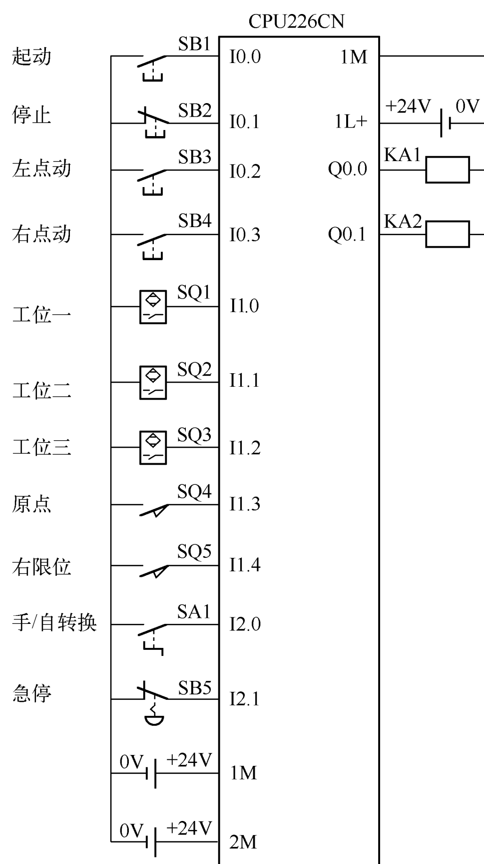


图 11-11　接线图

11.3.2　编写控制程序

程序如图 11-12 所示。

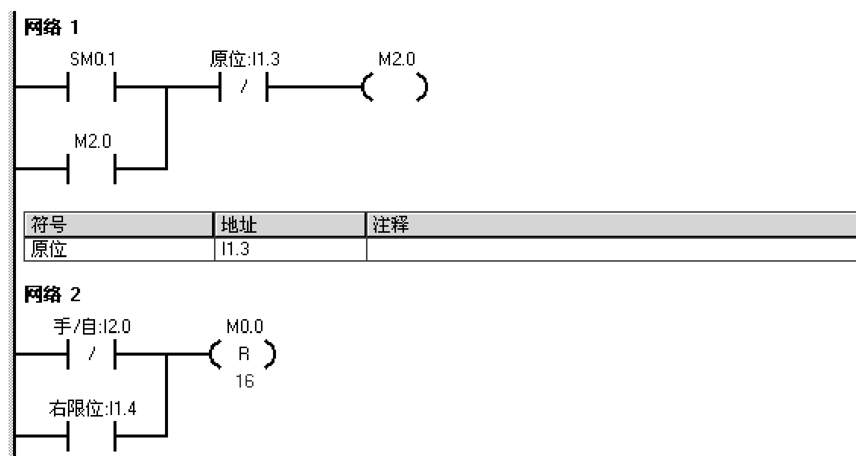
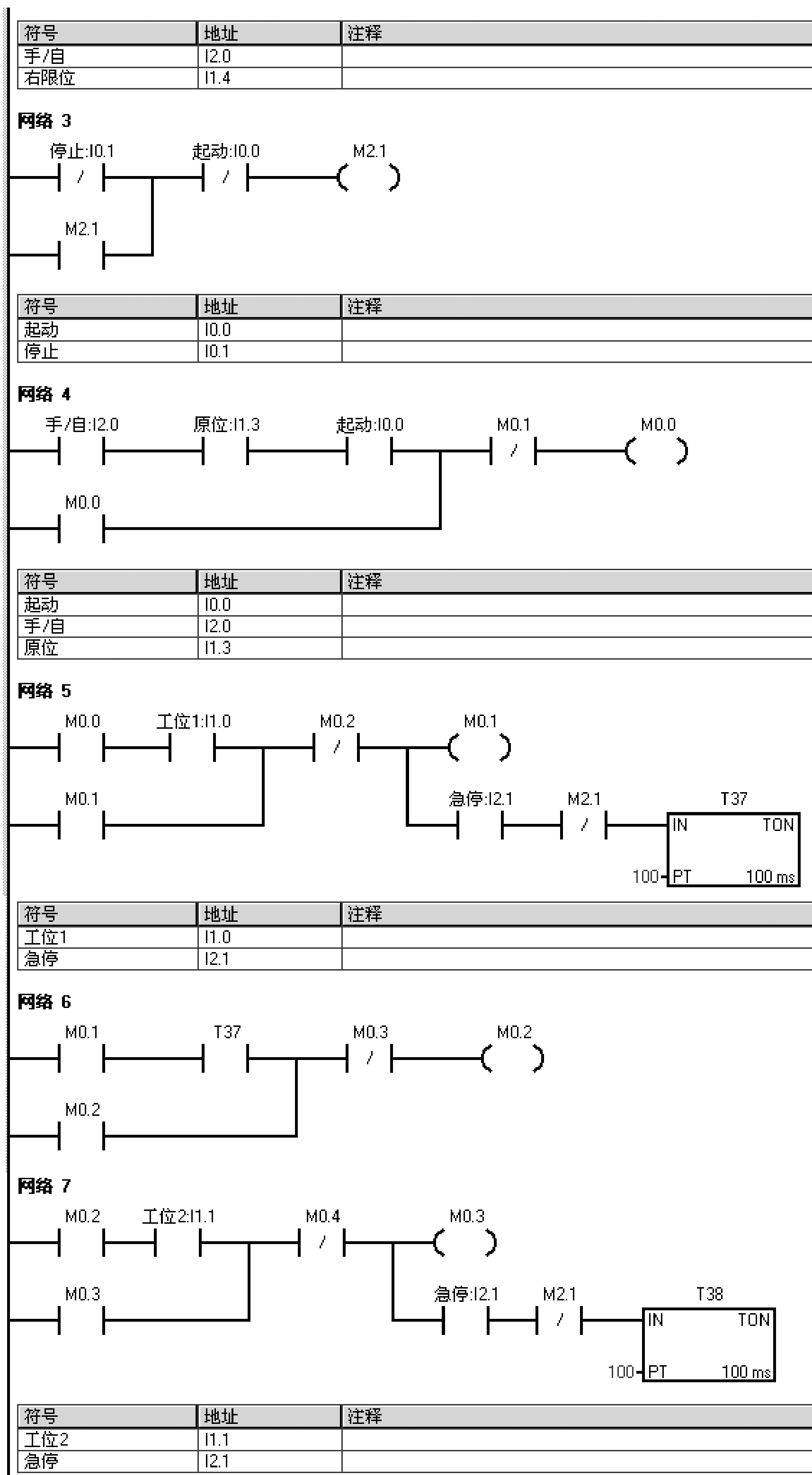


图 11-12　程序



符号	地址	注释
手/自	I2.0	
右限位	I1.4	

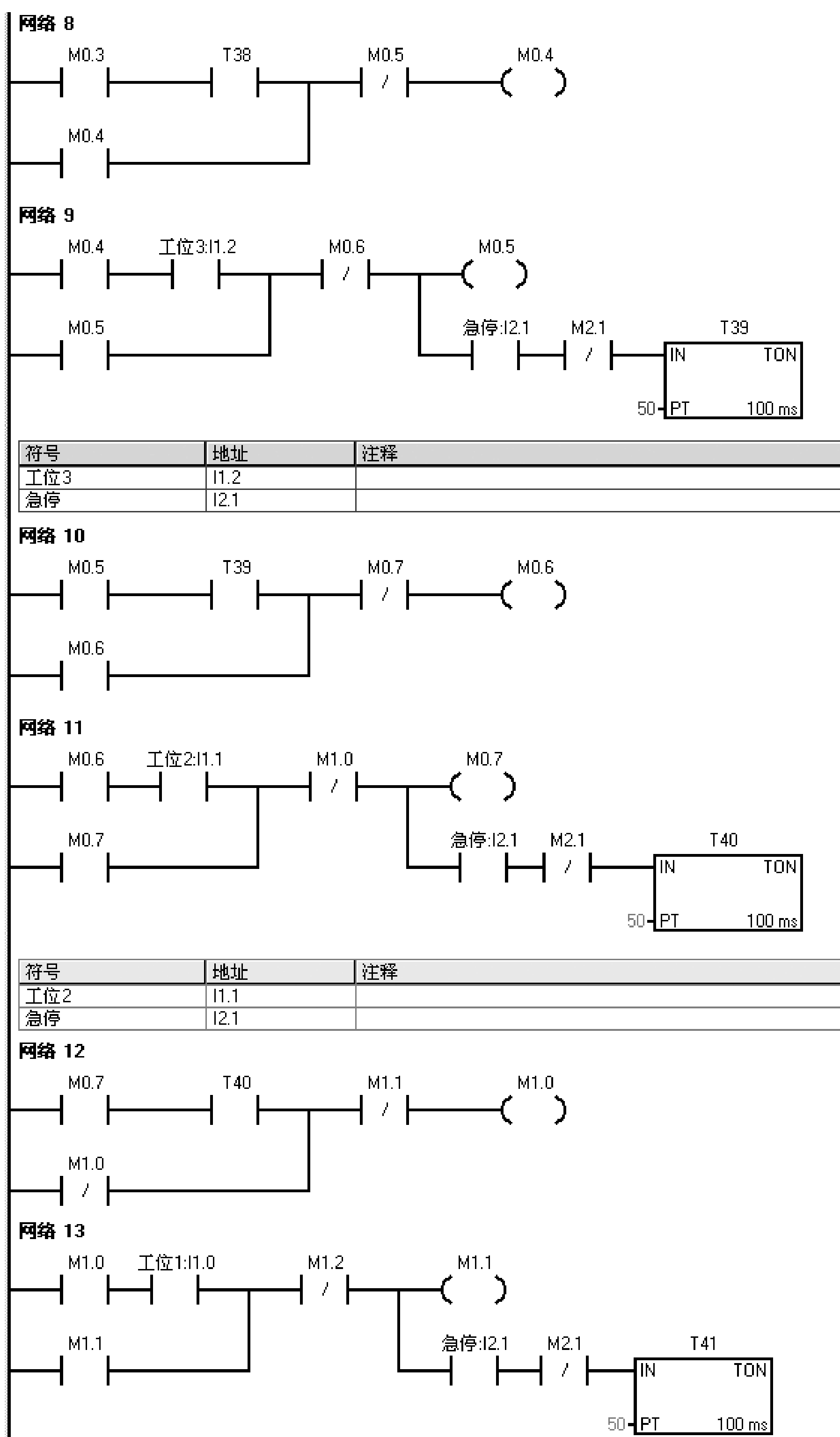
符号	地址	注释
起动	I0.0	
停止	I0.1	

符号	地址	注释
起动	I0.0	
手/自	I2.0	
原位	I1.3	

符号	地址	注释
工位1	I1.0	
急停	I2.1	

符号	地址	注释
工位2	I1.1	
急停	I2.1	

图 11-12　程序（续）



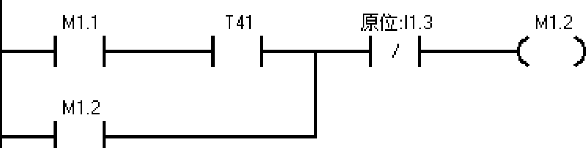
符号	地址	注释
工位3	I1.2	
急停	I2.1	

符号	地址	注释
工位2	I1.1	
急停	I2.1	

图 11-12　程序（续）

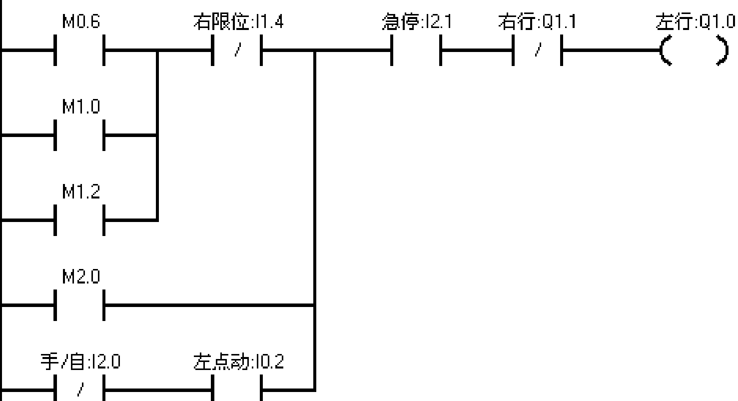
符号	地址	注释
工位1	I1.0	
急停	I2.1	

网络 14



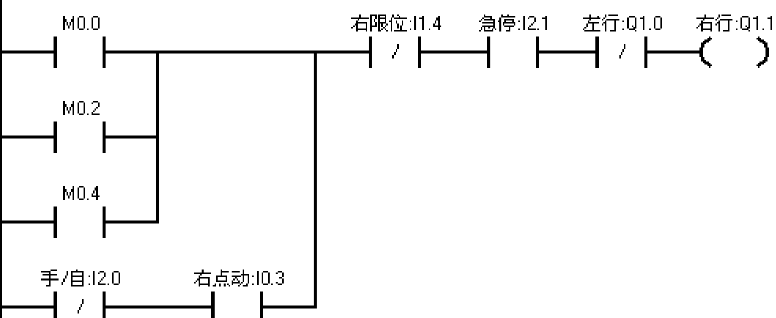
符号	地址	注释
原位	I1.3	

网络 15



符号	地址	注释
急停	I2.1	
手/自	I2.0	
右限位	I1.4	
右行	Q1.1	
左点动	I0.2	
左行	Q1.0	

网络 16



符号	地址	注释
急停	I2.1	
手/自	I2.0	
右点动	I0.3	
右限位	I1.4	
右行	Q1.1	
左行	Q1.0	

图 11-12　程序（续）

11.4 搬运站 PLC 控制系统

【**例 11-4**】搬运站的外形如图 11-13 所示，搬运站的下部是箱式底座，一方面起支撑作用，另一方面底座的箱子可作为控制柜，控制系统的 PLC 和电器元件都安装在内部；支架是铝合金型材，起支撑作用，主要用于固定回转气缸和导杆气缸，回转气缸可在 0°～180°范围内旋转，其输出转轴和导杆气缸相连，带动导杆气缸旋转；导杆气缸与普通气缸相比较，多了一个导向机构，以防止气缸活塞转动；手指气缸的作用是在搬运过程中起夹紧物体和释放物体的作用。很明显，搬运站有三个自由度，实际上是一个三自由度坐标式的机械手。

搬运站的功能就是将工件从上料检测站搬至加工站。当搬运站的 PLC 获得上料检测站的工件已经到位的信息，而且此时上料检测站处于空闲状态时。其控制过程如下：

控制系统先控制普通气缸上升到极限位置→导杆气缸缩回到极限位置→回转气缸旋转到左极限位置（以上的动作过程实际是系统回到原始位置）→导杆气缸升出到极限位置→普通气缸下行到下极限位置→手指气缸夹持物体→普通气缸上行到上极限位置→导杆气缸缩回→回转气缸旋转到右极限位置→导杆气缸伸出到极限位置→普通气缸下行到下极限位置→手指气缸释放物体，完成物体的搬运工作，一个工作循环完成。

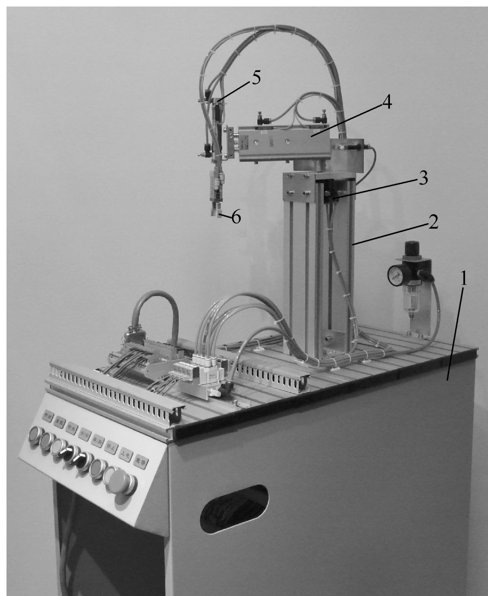


图 11-13　搬运站外形图

1—箱式底座　2—支架　3—回转气缸　4—导杆气缸　5—普通气缸　6—手指气缸

搬运站的气动原理图如图 11-14 所示。

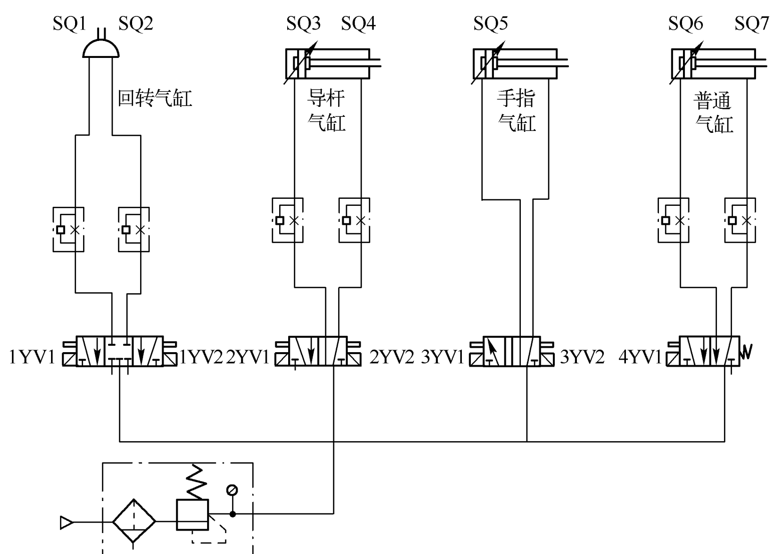


图 11-14　搬运站气动原理图

11.4.1　系统软硬件配置

1. 系统的软硬件

① 1 台 CPU 226CN。

② 1 套 STEP 7 - Micro/Win V4.0 SP9。

③ 1 根 PC/PPI 电缆。

2. PLC 的 I/O 分配

根据控制逻辑和图 11-14，对 PLC 进行 I/O 分配，见表 11-4。

表 11-4　搬运站 PLC 的 I/O 分配表

输入			输出		
名称	符号	输入点	名称	符号	输出点
开始按钮	SB1	I2.0	电磁阀	1YV1	Q0.0
复位按钮	SB2	I2.1	电磁阀	1YV2	Q0.1
停止按钮	SB3	I2.5	电磁阀	2YV1	Q0.2
左限位开关	SQ1	I0.0	电磁阀	2YV2	Q0.3
右限位开关	SQ2	I0.1	电磁阀	3YV1	Q0.4
内限位开关	SQ3	I0.2	电磁阀	3YV2	Q0.5
外限位开关	SQ4	I0.3	电磁阀	4YV1	Q0.6
夹紧限位开关	SQ5	I0.4	报警灯	HL1	Q1.5
上限位开关	SQ6	I0.5	信号灯	HL1	Q1.6
下限位开关	SQ7	I0.6	信号灯	HL2	Q1.7
检测物料限位开关	SQ8	I0.7			

3. 控制系统的接线

系统的接线图如图 11-15 所示。

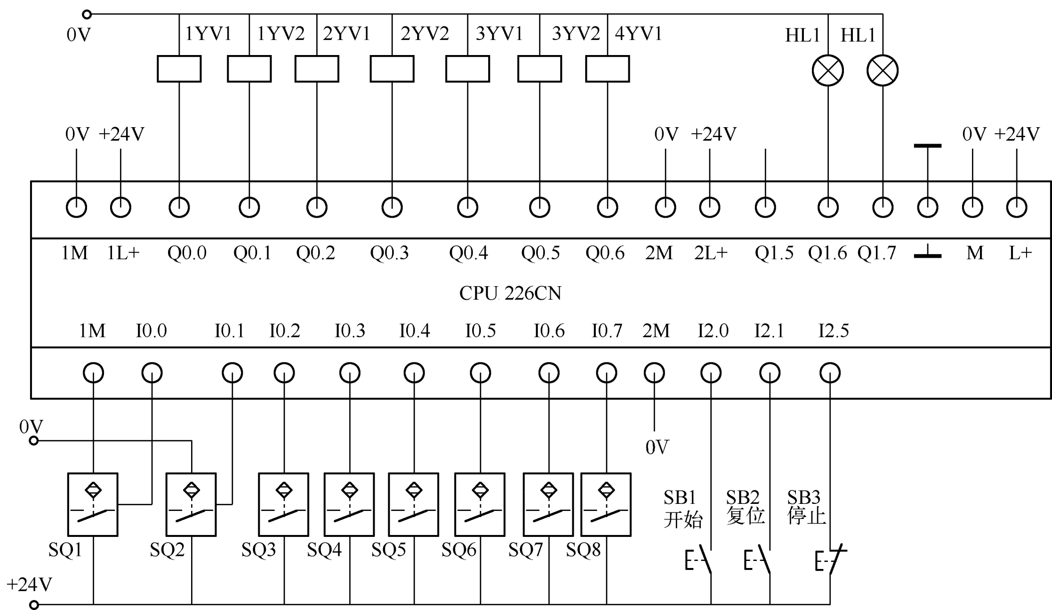


图 11-15 系统接线图

11.4.2 编写控制程序

先根据逻辑得出如图 11-16 所示的流程图，再编写程序，如图 11-17 所示。

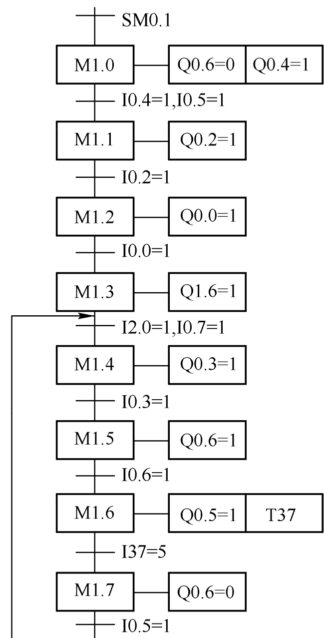


图 11-16 搬运站运行流程图

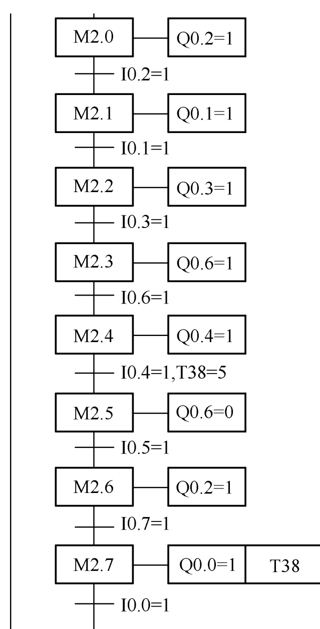


图 11−16 搬运站运行流程图（续）

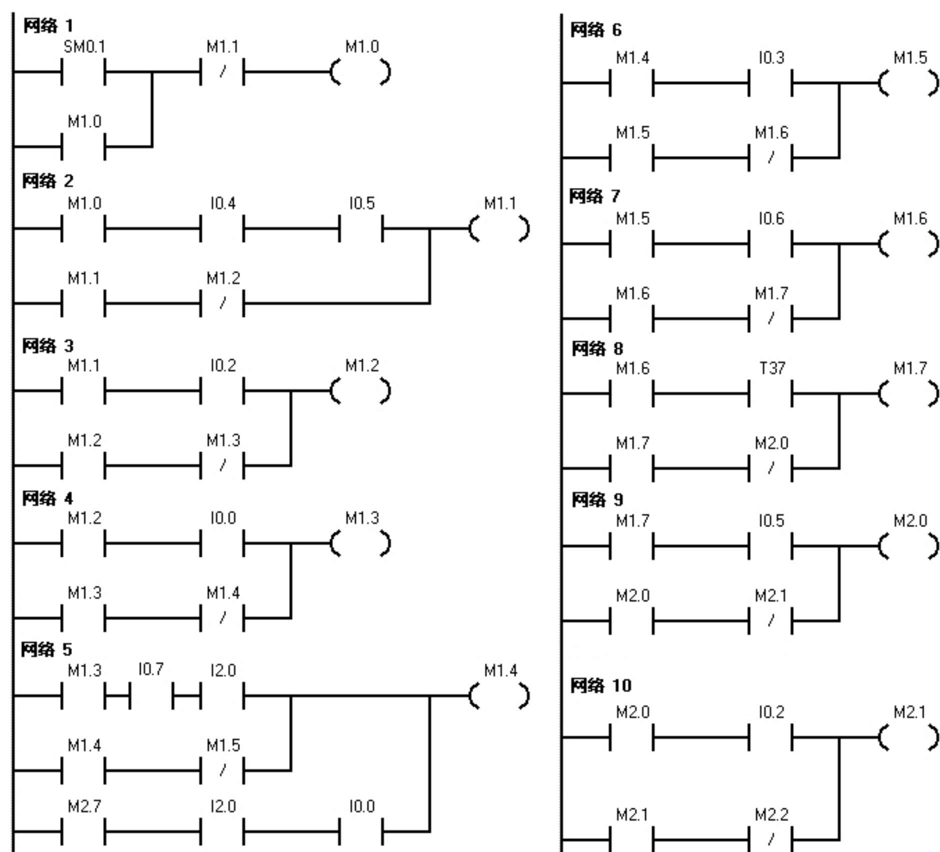


图 11−17 搬运站运行梯形图

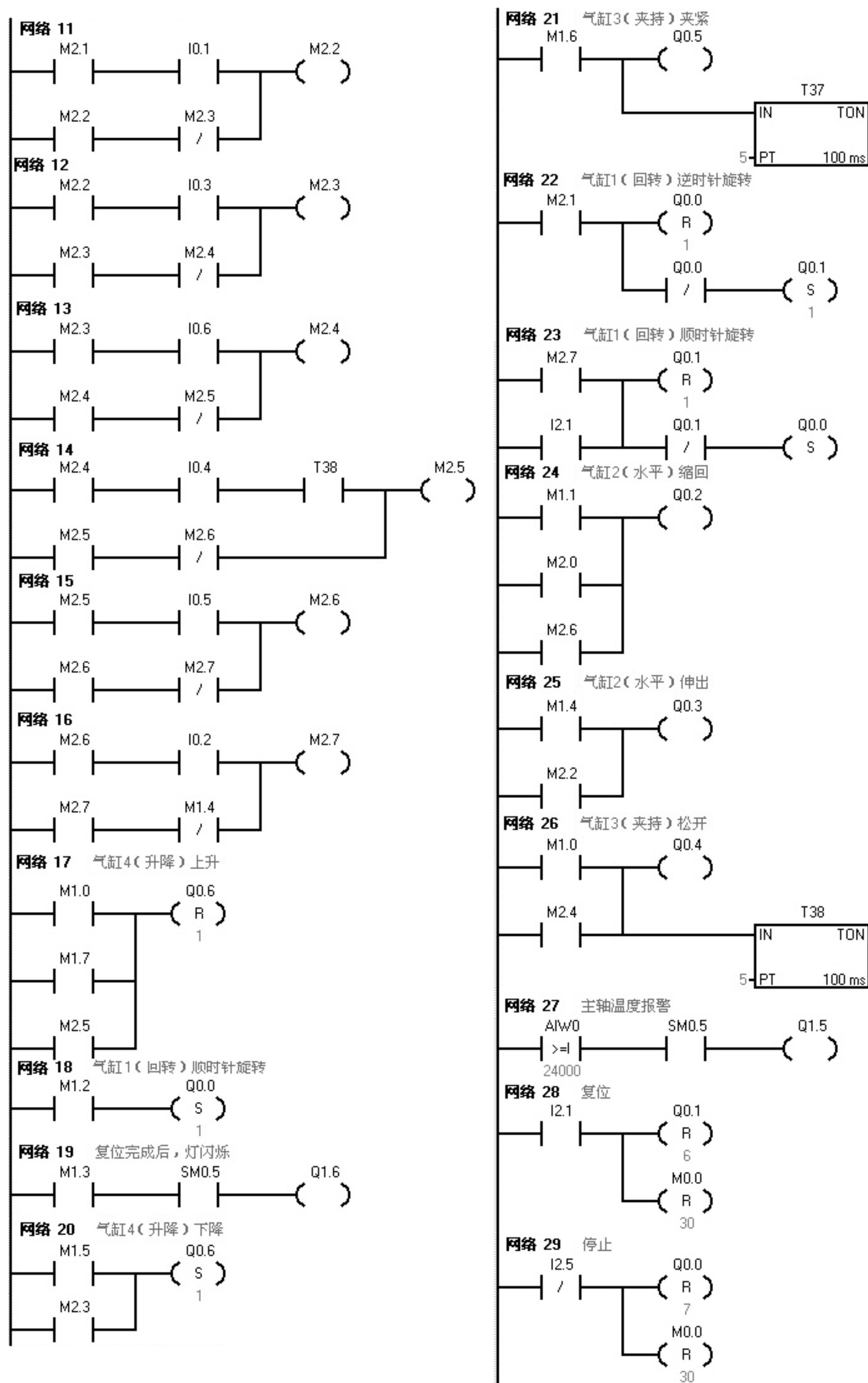


图 11-17　搬运站运行梯形图（续）

11.5　跳动度测试仪控制

【例 11-5】跳动度测试仪的机械结构比较简单，如图 11-18 所示，它主要由电动机 1、减速器 2、机架 3、主轴 4 和锁紧螺母 6 等组成。机架对整个系统起支撑作用，电动机和减速器为测试仪提供动力，锁紧螺母则将洗衣机内桶压紧到主轴上。机械的运动过程是：电动机接收到控制器的起动命令后，电动机转动，从而带动减速器旋转，减速器的输出轴与主轴相连，转动的主轴带动洗衣机内桶旋转。

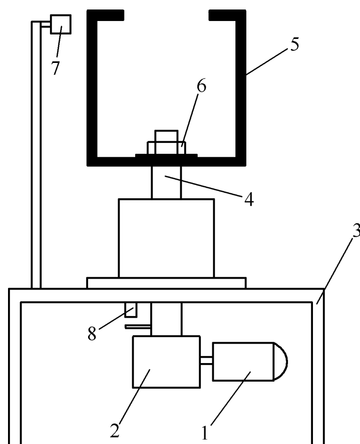


图 11-18　跳动度测试仪简图

1—电动机　2—减速器　3—机架　4—主轴　5—洗衣机内桶　6—锁紧螺母

7—激光位移传感器　8—接近开关

跳动度测试仪的工作原理：跳动度测试仪上有 1 个激光位移传感器（如图 11-18 所示的序号 7），主要实时采集激光传感器到内筒的距离。筒体转动一周，最大距离减去最小距离即是筒体的最大的跳动度。

跳动度测试仪的控制要求如下：

① 控制系统能够起动、停止和急停控制。

② 能够实现无级调速，调速范围是 90 ~ 450 r/min。

③ 当筒体转动一周时，测量一周的跳动度。

④ 实时测量电动机的转速。

11.5.1　系统软硬件配置

1. 系统的软硬件

① 1 台 CPU 222CN。

② 1 台 EM 235。

③ 1 套 STEP 7 - Micro/Win V4.0 SP9。

④ 1 根 PC/PPI 电缆。

2. PLC 的 I/O 分配

在 I/O 分配之前，先计算所需要的 I/O 点数，输入点为 5 个，输出点为 2 个，由于输入

输出最好留 15% 左右的余量备用，所以初步选择的 PLC 是 CPU 221CN 或者 CPU 222CN，又因为要使用扩展模块 EM 235CN，所以不能选择 CPU 221CN。西门子 S7－200 系列的 PLC，晶体管输出的类型要比继电器输出类型的 PLC 便宜一些，所以 PLC 最后定为 CPU 222CN（DC/DC/DC）。跳动度测试仪的 I/O 分配见表 11-5。

表 11-5　I/O 分配表

输　　入			输　　出		
名　　称	符　　号	输　入　点	名　　称	符　　号	输　出　点
起动按钮	SB1	I0.0	继电器	KA	Q0.0
停止按钮	SB2	I0.1			
测量按钮	SA1	I0.2			
光电开关	SQ1	I0.3			

3. 控制系统的接线

根据 I/O 分配表和题意，设计原理图如图 11-19 所示。SB5 是上电按钮，压下按钮 SB5 时，接触器 KM 线圈得电，接触器自锁，接触器常开触点闭合，电源与变频器、开关电源 VC 接通，进而开关电源为 CPU 222CN 和 EM 235 模块提供 DC ＋24 V 的电源。当压下按钮 SB4 时，切断了整个系统的电源。有紧急情况时，压下 SB3 按钮，也可以切断系统的电源。

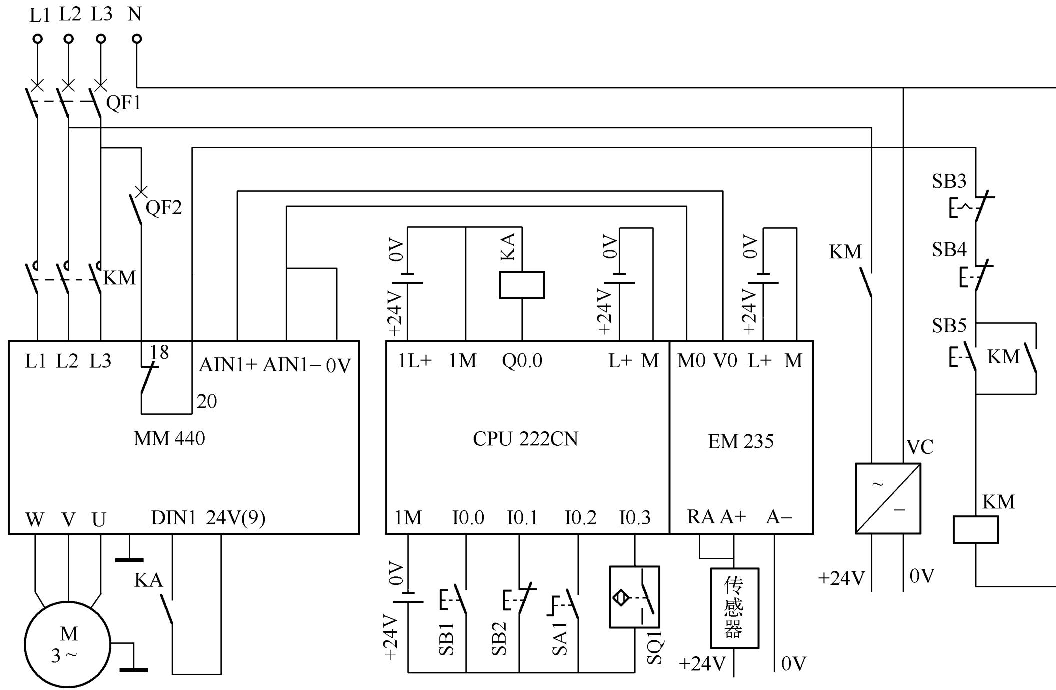


图 11-19　跳动度测试仪原理图

SB1 是起动按钮，可以使得变频器控制电动机转动；SB2 是停止按钮，可以使得变频器控制电动机停止转动；SB3 是测量按钮，当筒体运转平稳后，接通按钮，当光电开关第一次

有感应时，开始采集数据，当光电开关第二次有感应时，停止数据采集，表明系统采集了洗衣机筒体一周的数据，PLC 计算出跳动度，并显示在 HMI 上。

变频器上的 18 和 20 号端子，是变频器内部的继电器的常闭触点，当变频器处于故障和报警状态时，此常闭触点断开。

11.5.2 编写控制程序

数字量多段速度给定可以设定的速度段是有限的，而且不能做到无级调速，而外部模拟量输入可以做到无级调速，而且模拟量可以是电压信号或者电流信号，使用比较灵活，因此应用较广，但 S7－200 PLC（CPU 224XP 除外）输出模拟量时需要配置模拟量输出模块（如 EM 232），这增加了配置硬件的成本。

1. 系统接线

将 PLC、变频器、模拟量输出模块 EM 235 和电动机按照如图 11-19 所示接线。

2. 参数设定

查询 MM 440 变频器的说明书，再依次在变频器中设定表 11-6 中的参数。

表 11-6 变频器参数表

序　　号	变频器参数	出　厂　值	设　定　值	功 能 说 明
1	P0304	380	380	电动机的额定电压（380 V）
2	P0305	1.8	2.05	电动机的额定电流（2.05 A）
3	P0307	0.75	0.75	电动机的额定功率（0.75 W）
4	P0310	50	50	电动机的额定频率（50 Hz）
5	P0311	0	1 440	电动机的额定转速（1440 r/min）
6	P0700	2	2	选择命令源（由端子排输入）
7	P0756	0	0	选择 ADC 的类型（电压信号）
8	P1000	2	2	频率源（模拟量）
9	P701	1	1	数字量输入 1

3. 编写程序，并将程序下载到 PLC 中

电动机的转速为 1 440 r/min，减速器的减速比为 1∶3，要使电动机的转速为 90～450 r/min，变频器的频率要降低，降低到工频的 0.19～0.94 倍。又因我国的工频为 50 Hz，所以变频器的频率设定为 9.4～47 Hz。由于 0～32 000对应的输出模拟量值为 0～10 V，所以 3 200对应的模拟量为 1 V，电动机每转对应的数字量为 200/9，用一个表格说明则更加清楚，见表 11-7。默认变频器的频率为 10 Hz，可利用触摸屏设置不同的数值改变 VW0 的数值，从而达到调速的目的。

表 11-7 EM 235 模拟量模块和变频器对应关系

模拟量模块的数字量	模拟量模块 输出的模拟量	变频器的频率	电动机输出转速	减速器输出转速
0	0	0	0	0
6 000	1.9 V	9.4 Hz	270 r/min	90 r/min
30 000	0.94 V	47 Hz	1 350 r/min	450 r/min
32 000	10 V	50 Hz	1 440 r/min	480 r/min

激光位移传感器，输出的是 0～20 mA 的电流信号，其测量范围是 0～50 000 μm，所以位移为 50 000 μm 时对应的数字量是 32 000，或者说 25 μm 的位移对应的数字量是 16，这点在编写程序时要用到。

梯形图如图 11-20 所示。

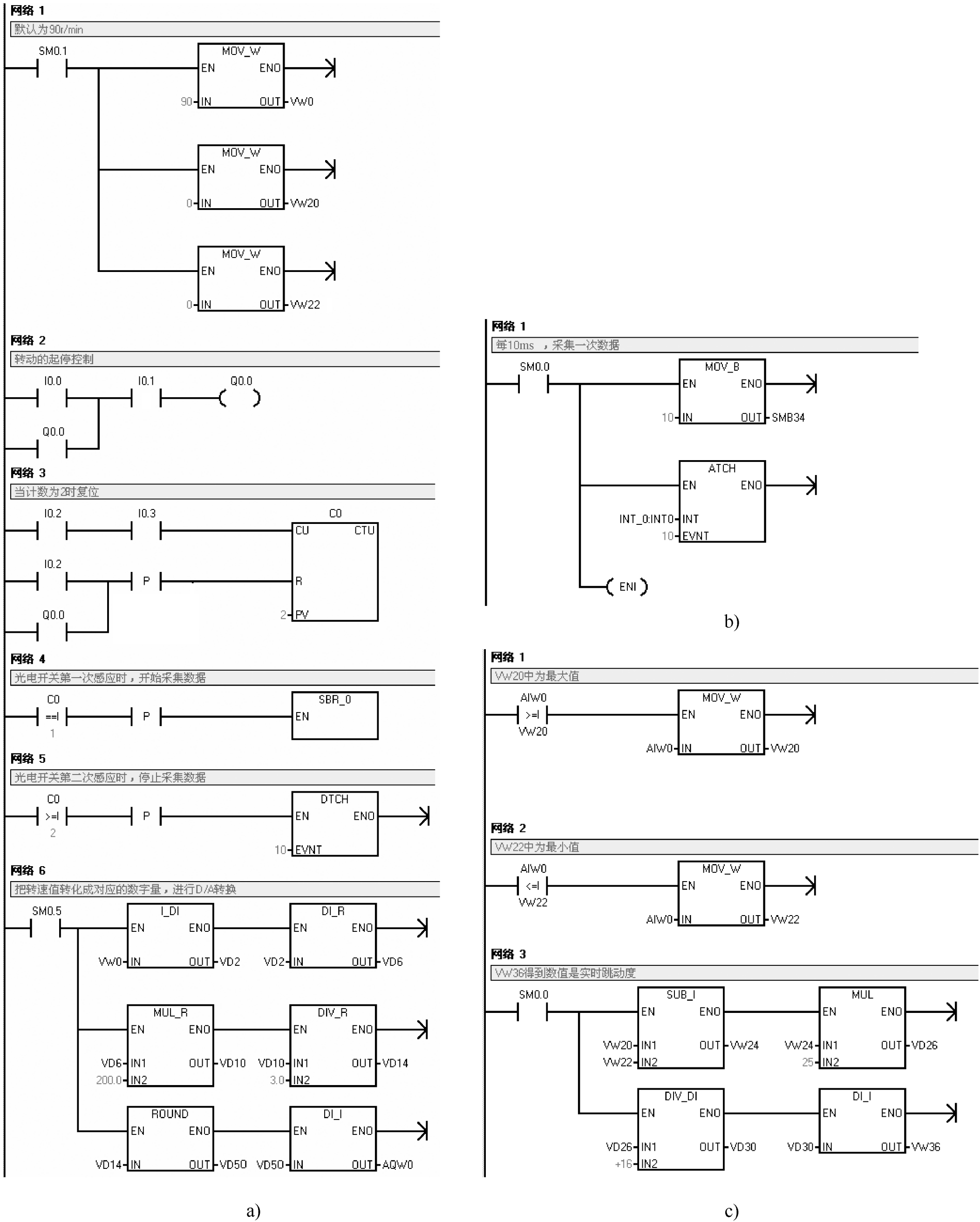


图 11-20　梯形图

a）主程序　b）子程序　c）中断程序

11.6 刨床 PLC 控制系统

【例 11–6】已知某刨床的控制系统主要由 PLC 和变频器组成，PLC 对变频器进行通信调速，变频器的运动曲线如图 11–21 所示，变频器以 20 Hz、30 Hz、50 Hz、0 Hz 和反向 50 Hz 运行，每种频率运行的时间都是 8 s，而且减速和加速时间都是 2 s（这个时间不包含在8 s 内），如此工作 2 个周期自动停止。要求如下：

① 设计此系统，画出原理图。

② 正确设置变频器的参数。

③ 编写程序。

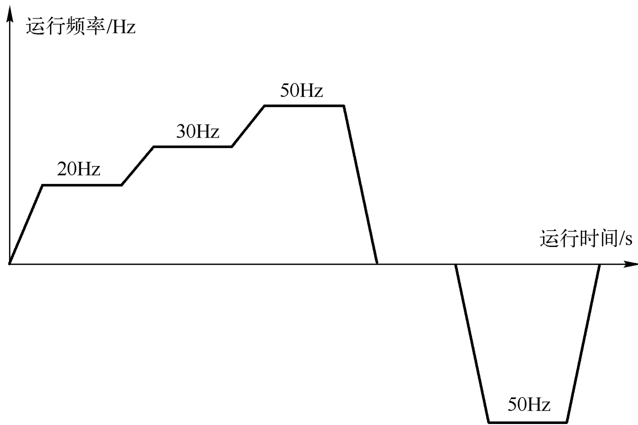


图 11–21　刨床的变频器的运动曲线

11.6.1　系统软硬件配置

1. 主要软硬件

① 1 台 CPU 226CN。

② 1 台 MM 440 变频器。

③ 1 套 STEP 7 – Micro/WIN V4.0 SP9。

④ 1 根 PC/PPI 电缆。

2. PLC 的 I/O 分配

PLC 的 I/O 分配见表 11–8。

表 11–8　PLC 的 I/O 分配表

名 称	符 号	输入点	名 称	符 号	输出点
起动按钮	SB1	I0.0	继电器		Q0.0
停止按钮	SB2	I0.1			
急停按钮	SB3	I0.2			

3. 控制系统的接线

控制系统的接线如图 11–22 所示。

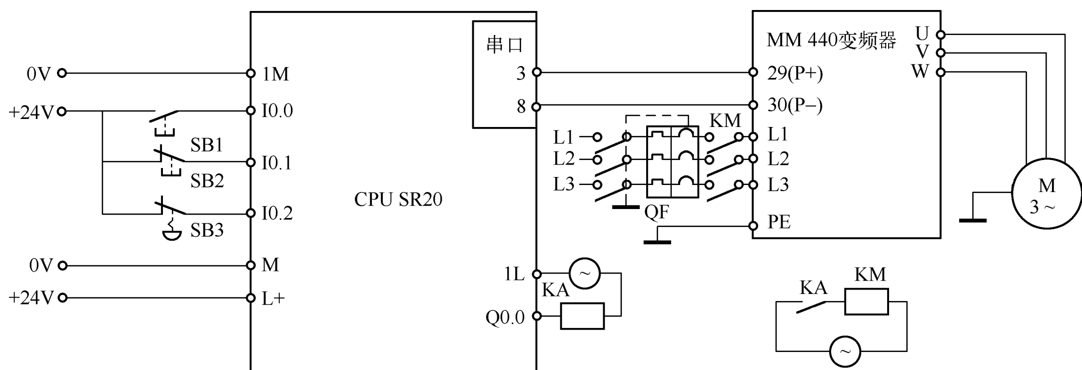


图 11–22　PLC 接线图

4. 变频器参数设定

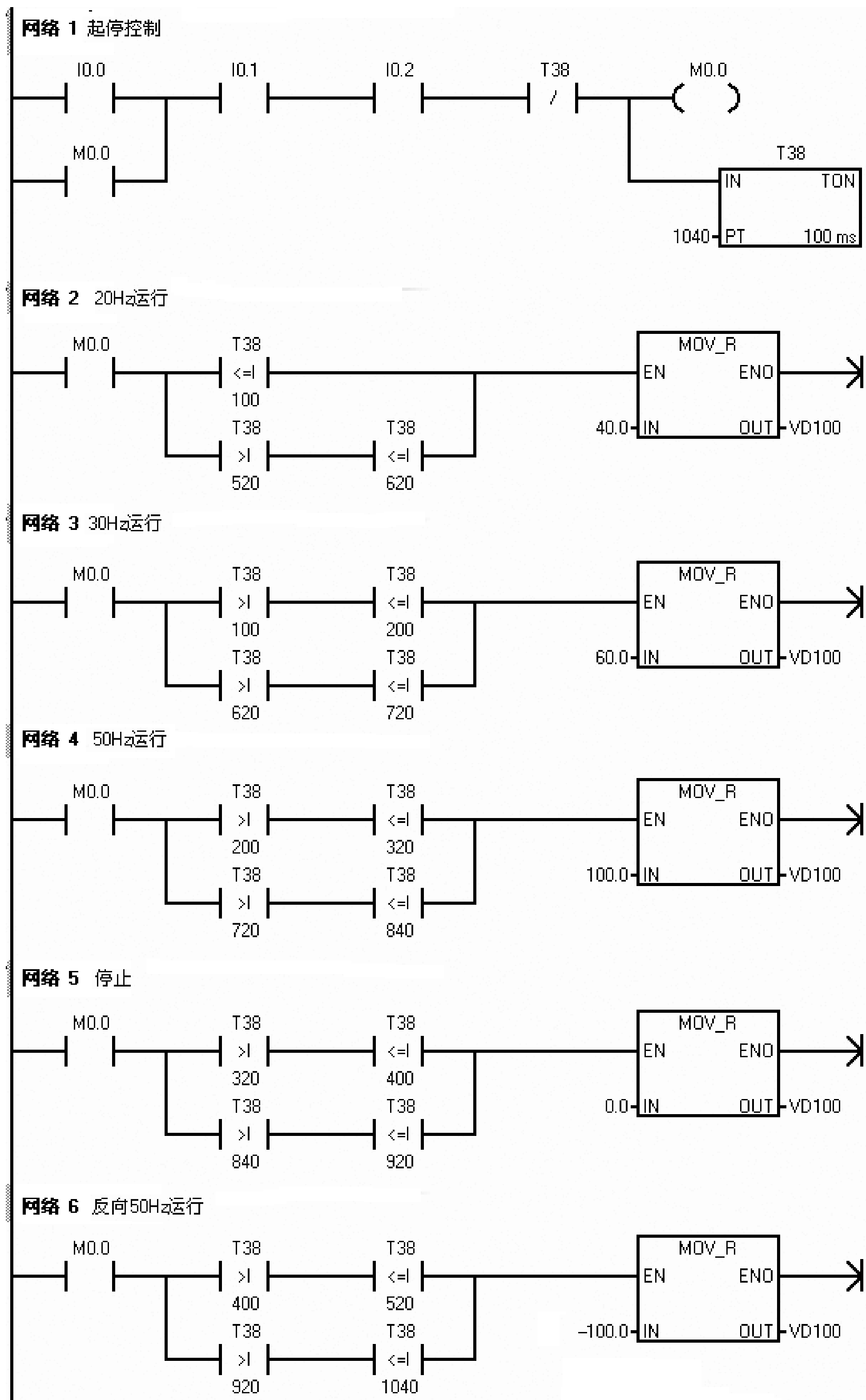
变频器的参数设定见表 11–9。

表 11–9　变频器的参数

序　　号	变频器参数	出　厂　值	设　定　值	功 能 说 明
1	P0005	21	21	显示频率值
2	P0304	380	380	电动机的额定电压（380 V）
3	P0305	19. 7	20	电动机的额定电流（0. 35 A）
4	P0307	7. 5	7. 5	电动机的额定功率（7. 5 kW）
5	P0310	50	50	电动机的额定频率（50 Hz）
6	P0311	0	1 440	电动机的额定转速（1 430 r/min）
7	P0700	2	5	选择命令源（COM 链路的 USS 设置）
8	P1000	2	5	频率源（COM 链路的 USS 设置）
9	P1000	10	2	斜坡上升时间
10	P1120	10	2	斜坡下降时间
11	P1121	6	6	USS 波特率（6 ~ 9 600）
12	P2011	0	0	站点的地址
13	P2012	2	2	PZD 长度
14	P2013	127	127	PKW 长度（长度可变）
15	P2014	0	0	看门狗时间

11. 6. 2　编写控制程序

从图 11–22 可见，一个周期的运行时间是 52 s，上升和下降时间直接设置在变频器中，也就是 P1120 = P1121 = 2 s，编写程序不用考虑。编写程序时，可以将 2 个周期当做一个周期考虑，编写程序更加方便。梯形图如图 11–23 所示。



网络 1 起停控制
I0.0
I0.1
I0.2
T38
M0.0
M0.0
T38
IN
TON
1040
PT
100 ms
网络 2 20Hz运行
M0.0
T38
<=I
100
T38
>I
520
T38
<=I
620
MOV_R
EN
ENO
40.0
IN
OUT
VD100
网络 3 30Hz运行
M0.0
T38
>I
100
T38
<=I
200
T38
>I
620
T38
<=I
720
MOV_R
EN
ENO
60.0
IN
OUT
VD100
网络 4 50Hz运行
M0.0
T38
>I
200
T38
<=I
320
T38
>I
720
T38
<=I
840
MOV_R
EN
ENO
100.0
IN
OUT
VD100
网络 5 停止
M0.0
T38
>I
320
T38
<=I
400
T38
>I
840
T38
<=I
920
MOV_R
EN
ENO
0.0
IN
OUT
VD100
网络 6 反向50Hz运行
M0.0
T38
>I
400
T38
<=I
520
T38
>I
920
T38
<=I
1040
MOV_R
EN
ENO
-100.0
IN
OUT
VD100

图 11-23　梯形图

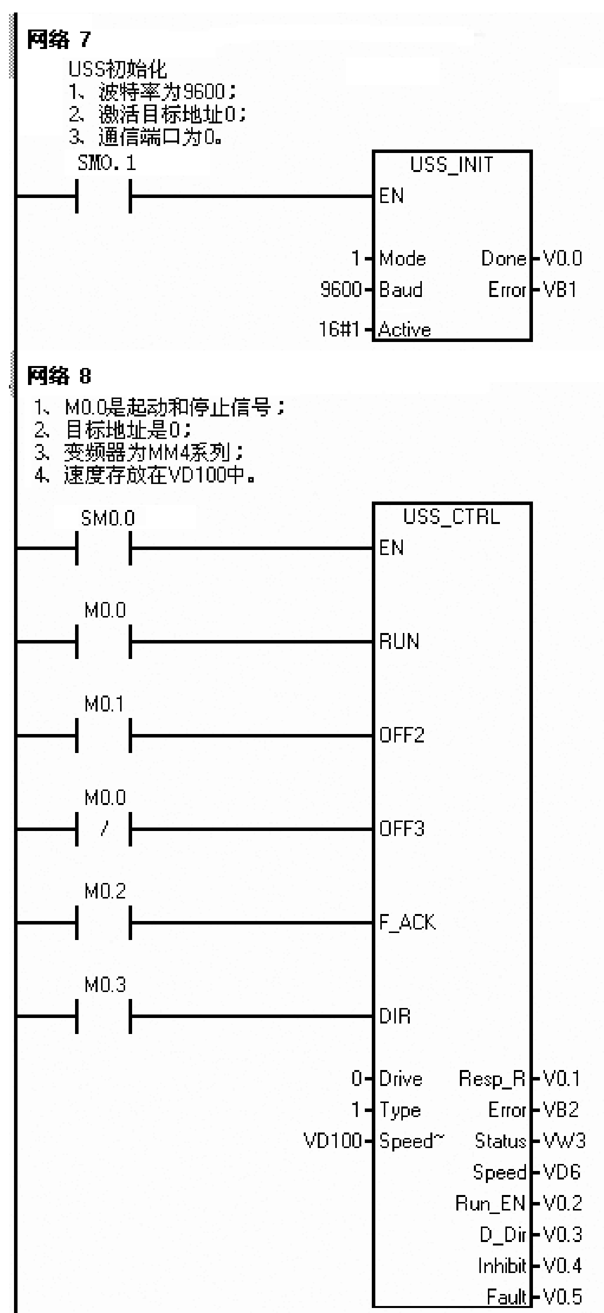


图 11-23　梯形图（续）

11.7　电镀生产线的 PLC 控制

【例 11-7】有一条电镀生产线，结构示意图如图 11-24 所示，有 4 个工位，气动机械手在 A 位置抓取工件后到 B、C、D 共 4 个工位，在每个工位停留时间是 1 s，A、B、C、D，

相邻 2 个工位中心之间的距离是 400 mm，机械手的移动有滚珠丝杠带动，滚珠丝杠的螺距是 2 mm，直流电动机直接与光电编码器和丝杠相连，光电编码器为 1024 线。机械手上有 2 个气缸，一个用于上升和下降的升降气缸，另一个用于抓取工件的手指气缸。系统的控制要求如下：

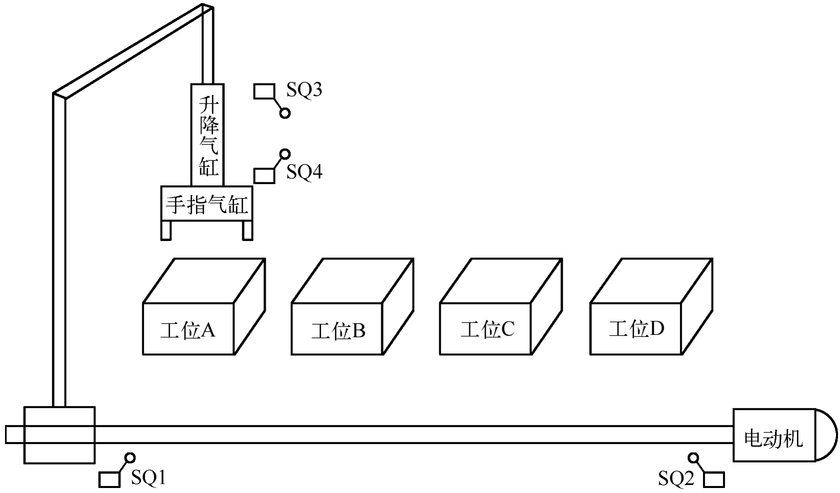


图 11-24　结构示意图

① 有手动和自动模式，在手动模式时，有上升、下降、右行和左行 4 个按钮用于调试。

② 有复位按钮，当按下“复位”按钮时，气缸自动上行到上极限位置，电动机拖动机械手到左极限位置，复位完成，有指示灯闪亮，提示操作者。

③ 设有暂停和急停按钮，压下“暂停”按钮，完成一个工作循环，机械手停止工作，而压下“急停”按钮，系统立即停止工作。

④ 当处于自动模式时，在初始位置 A 时，按下“起动”按钮时，工作过程如下：

手指气缸松开→升降气缸下降到下极限位置→手指气缸抓取工件，延时 1 s→机械手移动到工位 B，延时 2 s→机械手移动到工位 C，延时 2 s→机械手移动到工位 D，延时 2 s→机械手移动到工位 A，释放工件，延时 1 s→升降气缸上升到上极限位置，完成一个工作周期，之后如此往复循环。

11.7.1　系统软硬件配置

1. 主要软硬件配置

① 1 台 CPU 226CN。

② 1 台 光电编码器。

③ 1 套 STEP 7 - Micro/WIN V4.0 SP9。

④ 1 根 PC/PPI 电缆。

2. PLC 的 I/O 分配

PLC 的 I/O 分配见表 11-10。

表 11-10　PLC 的 I/O 分配表

输　　入			输　　出		
名　　称	符　　号	输　入　点	名　　称	符　　号	输　出　点
	I0.0	高速输入	YV1	Q0.0	夹紧
SB1	I0.1	复位	YV2	Q0.1	上行
SA1	I0.2	手动/自动转换	YV3	Q0.2	下行
SB2	I0.3	起动	YV4	Q0.3	左行
SB3	I0.4	停止	YV5	Q0.4	右行
SB4	I0.5	急停	HL1	Q0.5	信号灯
SQ3	I0.6	上限位			
SQ4	I0.7	下限位			
SQ1	I1.0	左限位			
SQ2	I1.1	右限位			
SB5	I1.2	手动上行			
SB6	I1.3	手动下行			
SB7	I1.4	手动左行			
SB8	I1.5	手动右行			
SB9	I1.6	手动夹紧			

3. 控制系统的接线

控制系统的接线如图 11-25 所示。

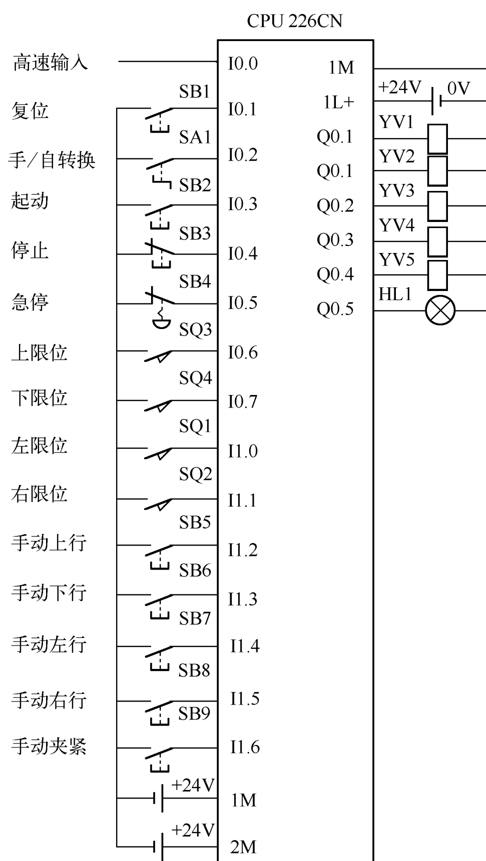


图 11-25　控制系统的接线

11.7.2 编写控制程序

编写程序如图 11-26 所示。

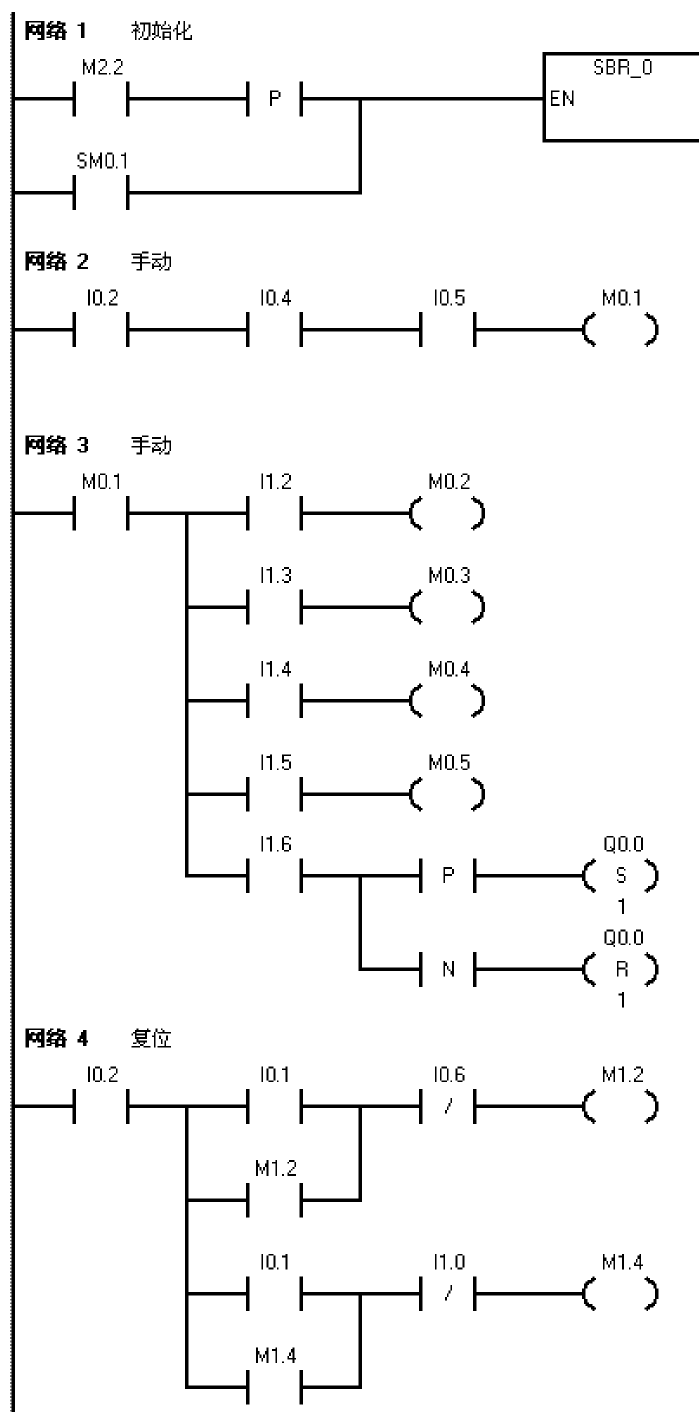


图 11-26 程序

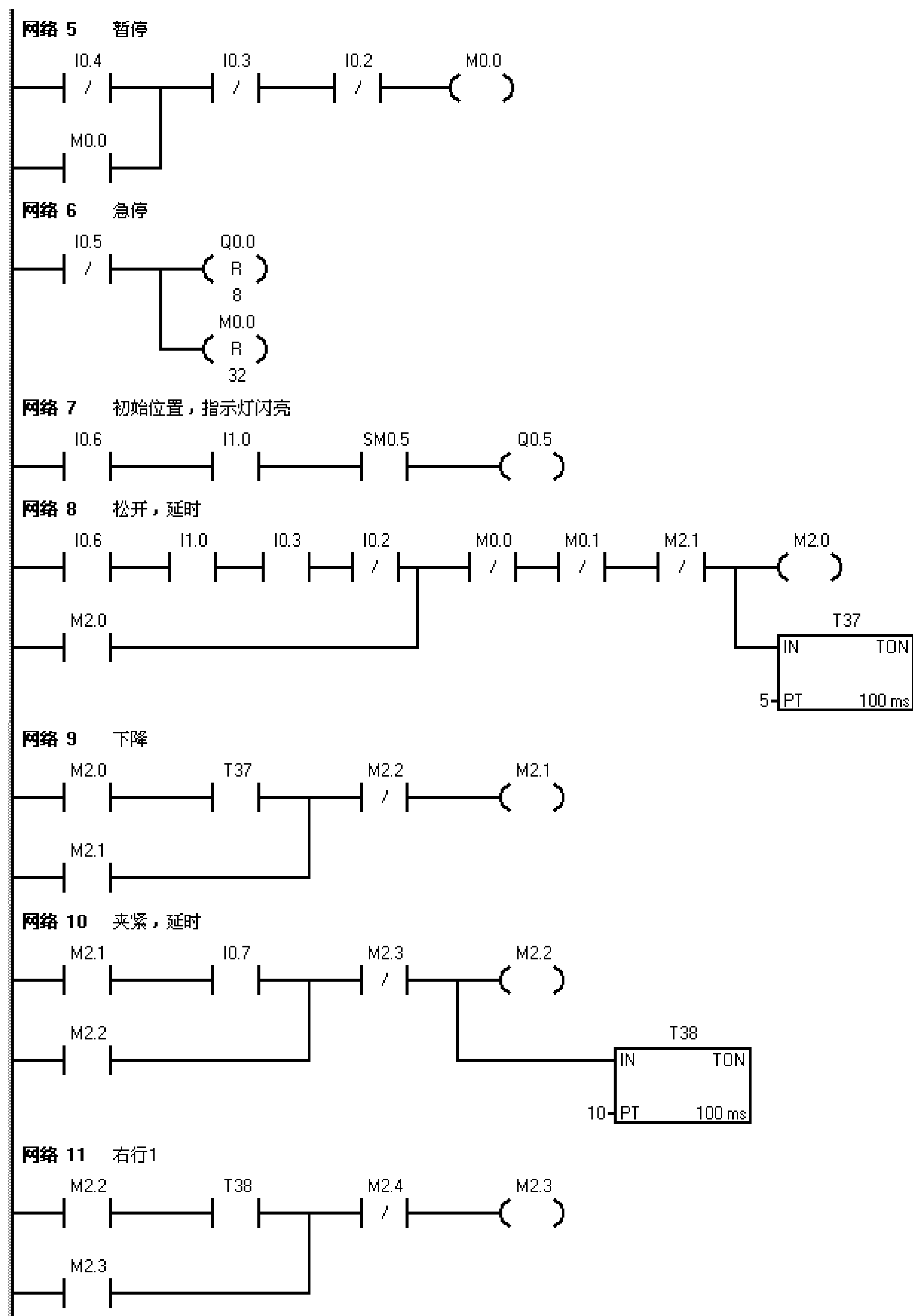

网络 5 暂停
I0.4
I0.3
I0.2
M0.0
M0.0
网络 6 急停
I0.5
Q0.0
R
8
M0.0
R
32
网络 7 初始位置，指示灯闪亮
I0.6
I1.0
SM0.5
Q0.5
网络 8 松开，延时
I0.6
I1.0
I0.3
I0.2
M0.0
M0.1
M2.1
M2.0
M2.0
T37
IN
TON
5
PT
100 ms
网络 9 下降
M2.0
T37
M2.2
M2.1
M2.1
网络 10 夹紧，延时
M2.1
I0.7
M2.3
M2.2
M2.2
T38
IN
TON
10
PT
100 ms
网络 11 右行1
M2.2
T38
M2.4
M2.3
M2.3

图 11-26 程序（续）

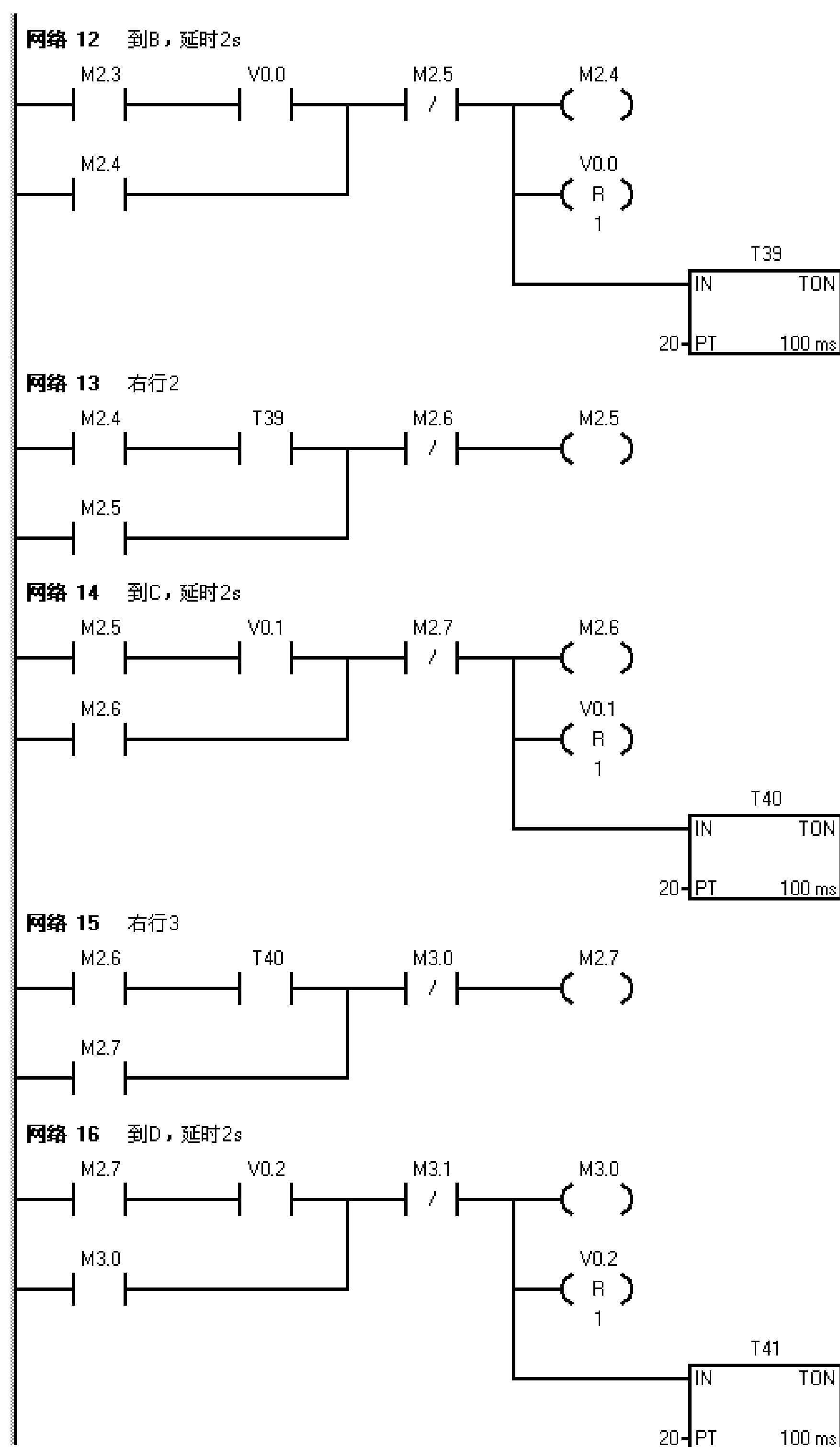
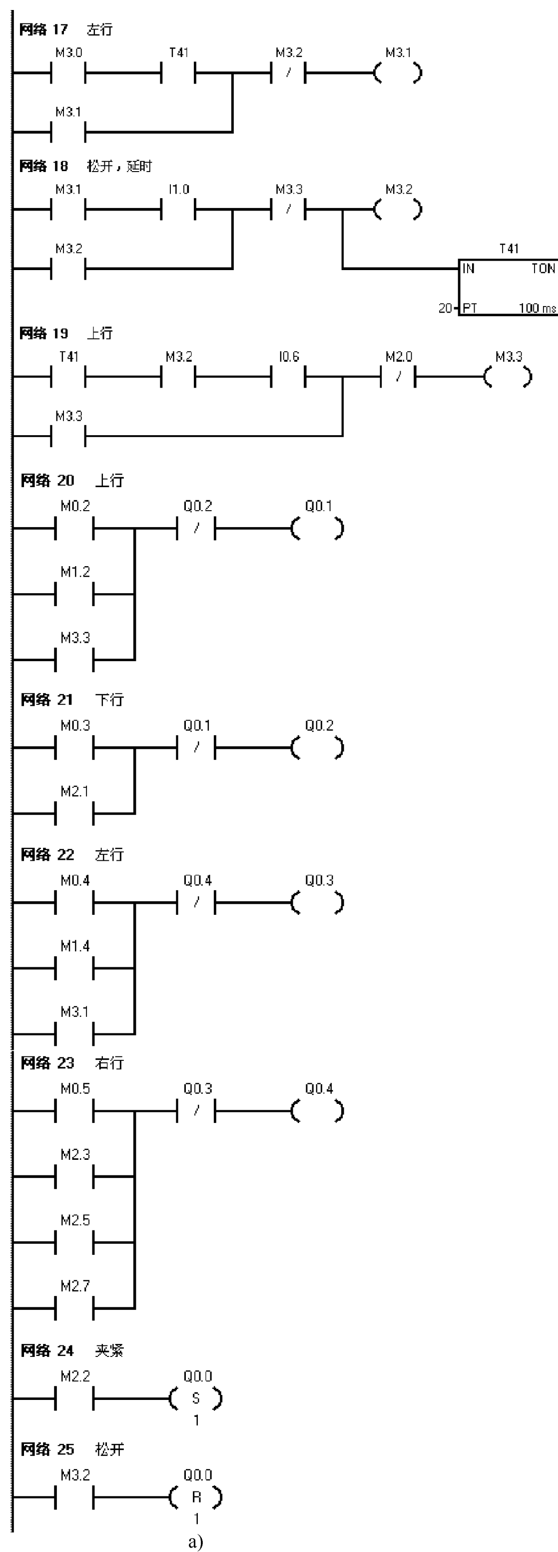

网络 12　到B，延时2s
M2.3
V0.0
M2.5
M2.4
M2.4
V0.0
R
1
T39
IN
TON
20
PT
100 ms
网络 13　右行2
M2.4
T39
M2.6
M2.5
M2.5
网络 14　到C，延时2s
M2.5
V0.1
M2.7
M2.6
M2.6
V0.1
R
1
T40
IN
TON
20
PT
100 ms
网络 15　右行3
M2.6
T40
M3.0
M2.7
M2.7
网络 16　到D，延时2s
M2.7
V0.2
M3.1
M3.0
M3.0
V0.2
R
1
T41
IN
TON
20
PT
100 ms

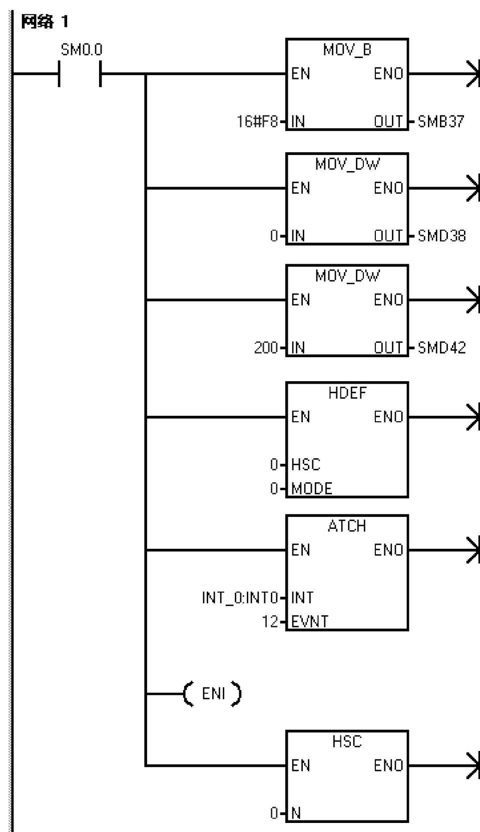
图 11-26　程序（续）



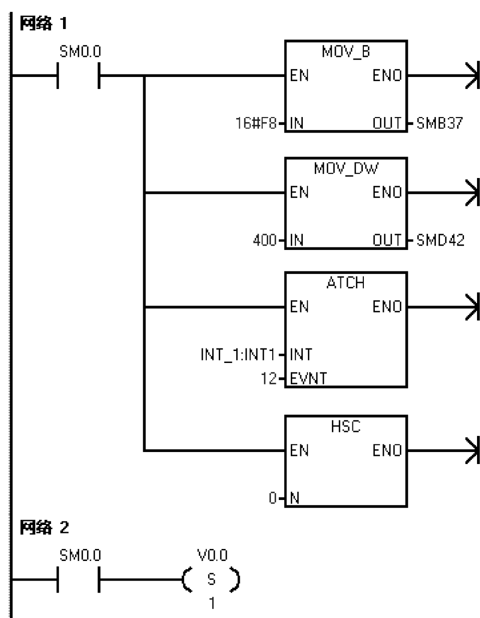
a)

图 11-26 程序（续）

a）主程序



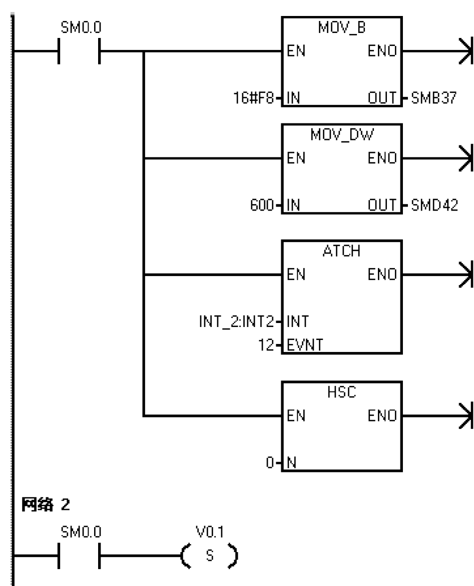
b)



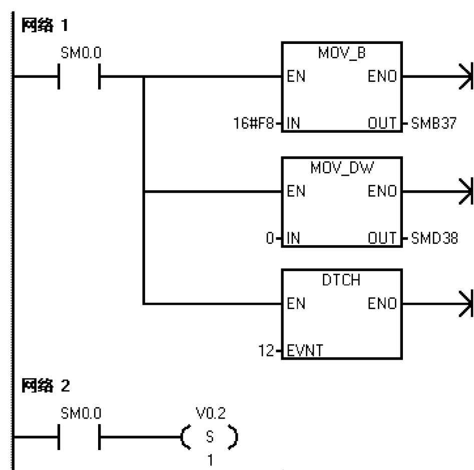
c)

图 11-26　程序（续）

b）子程序　c）中断程序 INT_0



d)



e)

图 11-26　程序（续）

d）中断程序 INT_1　e）中断程序 INT_2

11.8　十字滑台的 PLC 控制

【例 11-8】 十字滑台应用面广，可广泛应用于各种钻、铣、镗类组合机床及专用机床。一般在十字滑台上安装工件或动力头等相关附件后，通过十字滑台的进给运动，对工件进行铣削、钻削和镗削加工。用多个不同规格的滑台，然后在相应的滑台上面装上动力头和工件组合在一起就形成了组合机床，可进行复杂零部件的加工或进行批量生产，大大提高工件的生产率和制造精度。

十字滑台的外形如图 11-27 所示。十字滑台实际上就是两套相互垂直的滑台组成，通常将一个滑台当做 X 轴，另一个则当做 Y 轴。步进电动机通过联轴器和滚珠丝杠连接在一

起，步进电动机转动从而带动滑板在导轨上滑动。滑台的 X 轴和 Y 轴的极限位置上各有两个限位开关，X 轴和 Y 轴起始位置的限位开关可以作为坐标轴的原点。十字滑台的滚珠丝杠长 180 mm。

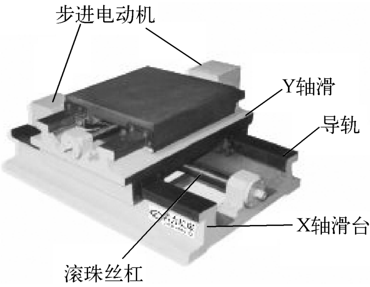


图 11-27 十字滑台外形图

从理论上看，通过控制 X 和 Y 方向的滑台的进给，能使滑台上的工件或者刀具等到达平面上的任何一点。工业上实际应用的十字滑台，以伺服系统驱动的最为常见。十字滑台的控制要求如下：

十字滑台有 X 和 Y 两个方向，控制系统的每个方向有停止、复位、前进和后退功能，前进和后退的位移由人机界面或者直接在程序中设定。每个方向都要有限位，即 X 和 Y 方向各有两个极限位开关。

11.8.1 系统软硬件配置

1. 系统软硬件

① 1 台 CPU 226CN。

② 2 套 步进驱动系统。

③ 1 套 STEP 7 - Micro/WIN V4.0 SP9。

④ 1 根 PC/PPI 电缆。

2. PLC 的 I/O 分配

在 I/O 分配之前，先计算所需要的 I/O 点数，由于输入输出最好留 15% 左右的余量备用（Y 方向未画），所以初步选择的 PLC 是 CPU 224CN。又因为要使用 PLC 的高速输出点，所以 PLC 最后定为 CPU 224CN（DC/DC/DC）。十字滑台的 I/O 分配见表 11-11。

表 11-11 I/O 分配表

输入			输出		
名称	符号	输入点	名称	符号	输出点
X 方向起动按钮	SB1	I0.0	X 方向高速输出		Q0.0
X 方向停止按钮	SB2	I0.1	X 方向控制		Q0.2
复位按钮	SB3	I0.2	复位完成标志	HL	Q0.3
X 方向前进按钮	SB4	I0.3			
X 方向后退按钮	SB5	I0.4			
X 方向限位 1	SQ1	I1.0			
X 方向限位 2	SQ2	I1.1			
X 原点	SQ3	I1.2			

3. 设计电气原理图

根据 I/O 分配表和题意，设计原理图如图 11-28 所示的十字滑台。SB2 是停止按钮，压下 SB2 按钮时，X 方向的运动停止。

【关键点】 2 个极限位开关应选用常闭触点的接近开关，这点很重要。X 前进和 X 后退都是点动，限位开关对点动不起限位作用。

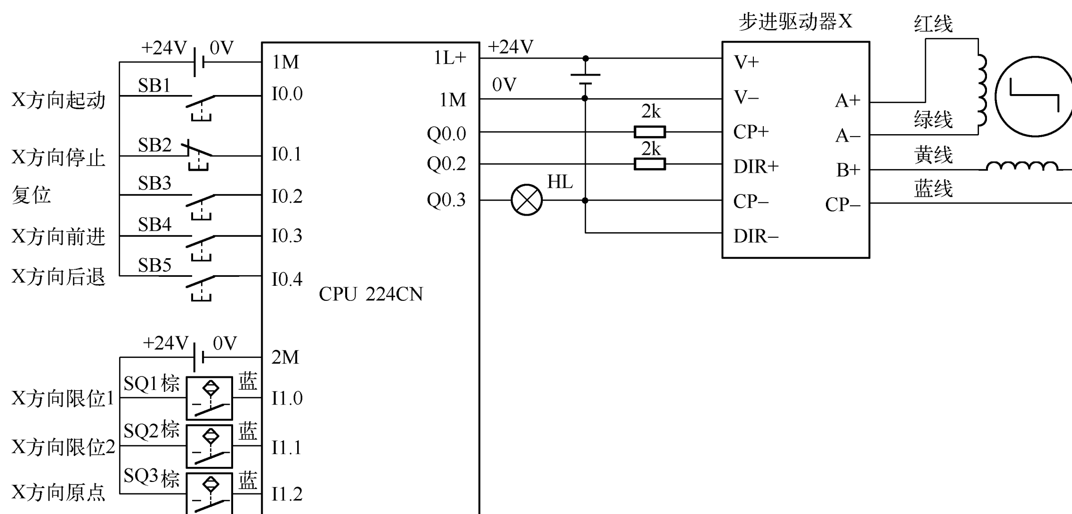


图 11-28　十字滑台原理图

11.8.2　编写控制程序

1. 复位的含义

十字滑台复位实际上就是回到原点，回原点的方式有几种，回原点方式的选择可据实际情况而定，以下介绍一种简单回原点的方式，具体如下：

当压下“复位”按钮时，X 方向的步进电动机拖动十字滑台以较快的速度后退（向原点方向），当接近开关 SQ3 检测到滑台的 A 边时，步进电动机以较慢的速度后退，直到滑台的 B 边离开原点的接近开关 SQ3，步进电动机停转，复位完成。Y 方向的复位也是类似的。复位（回原点）示意图如图 11-29 所示。

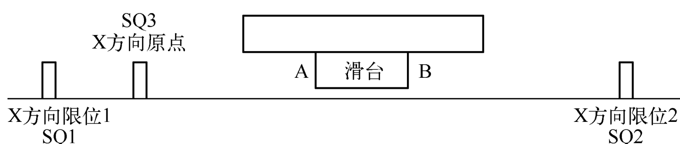


图 11-29　复位示意图

【关键点】 减速停机就是为了保证回原点的准确性。

2. 相关计算

已知步进电动机的步距角是 1.8°，所谓步距角就是步进电动机每接收到一个脉冲信号后，步进电动机转动的角度。步进电动机和滚珠丝杠直接相连，滚珠丝杠的螺距为 2 mm，也就是说步进电动机每转一圈，工作台水平移动 2 mm。

假设程序中要求步进电动机 X 方向前进 20 mm，转速是 400 r/min，那么程序中的特殊寄存器 SMW68 和 SMD72 如何设置？

1）对于初学者而言，这个计算的确有点麻烦，先计算脉冲数 n。由于前进的位移是 20 mm，则需要步进电动机转动的圈数为 20/2 = 10。电动机转动 10 圈，需要接收的脉冲数为

$$n = 10 \times \frac{360^{\circ}}{1.8^{\circ}} = 2\ 000$$

所以，SMD72 赋值为 2 000。

2）速度的计算相对麻烦，SMW68 中存放的是脉冲的周期，一般其单位是 μs。

$$T = 10^{6} \times \frac{60}{400 \times \frac{360^{\circ}}{1.8^{\circ}}}\ \mu s = 750\ \mu s$$

即每秒发出 1 333 个脉冲，这个数值在后面要用到。

3. 编写程序

主程序如图 11-30 所示，子程序如图 11-31 所示。

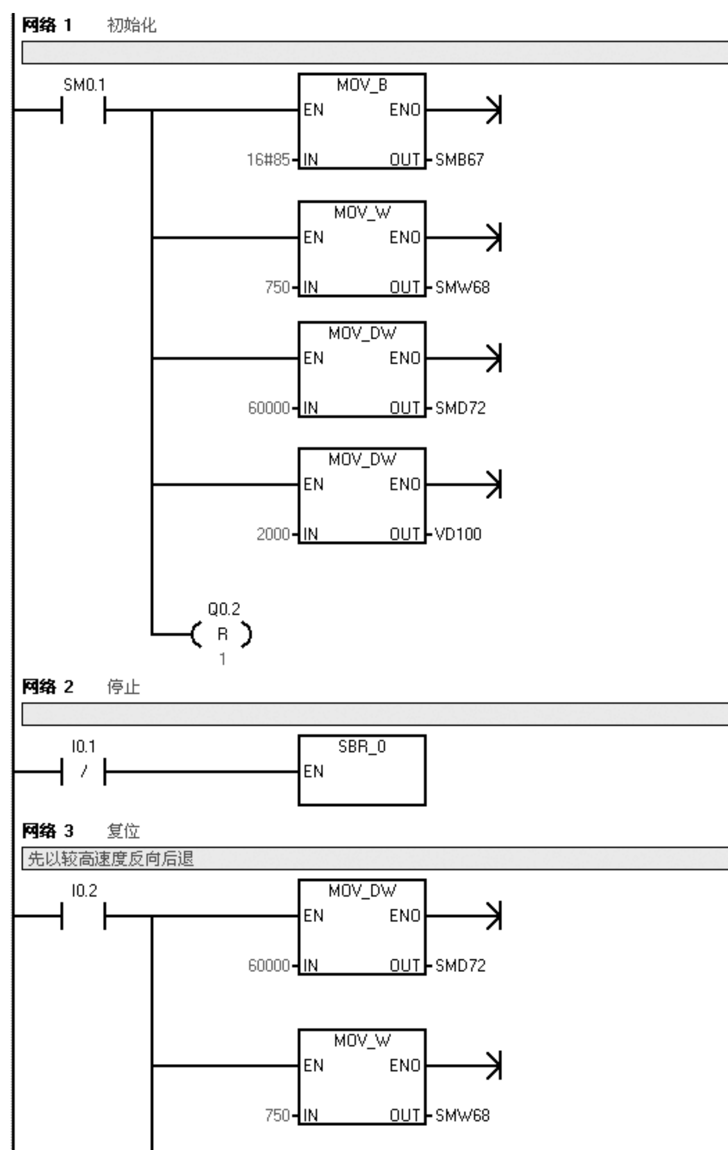


图 11-30　子程序

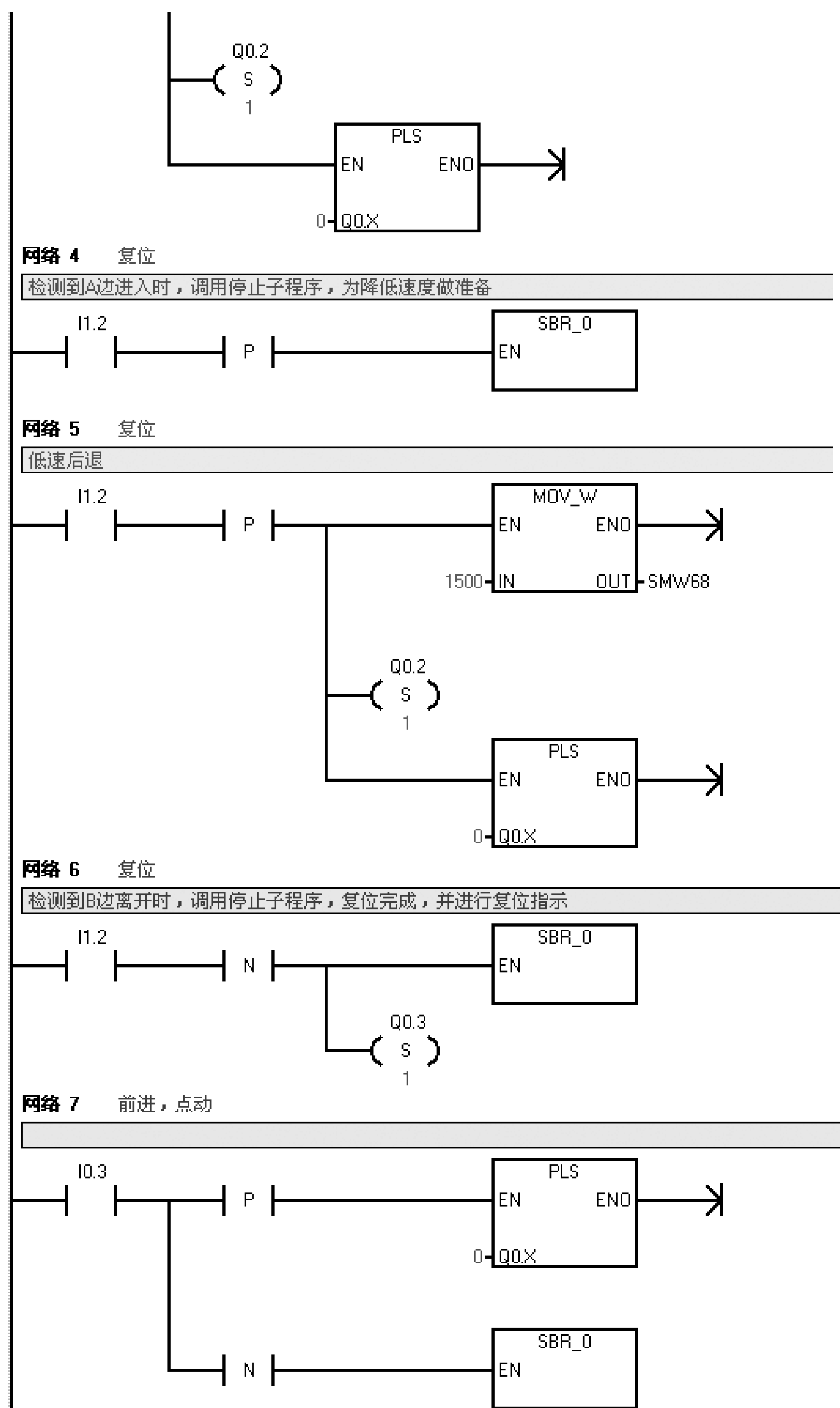

Q0.2
S
1
PLS
EN
ENO
0
Q0.X
网络 4 复位
检测到A边进入时，调用停止子程序，为降低速度做准备
I1.2
P
SBR_0
EN
网络 5 复位
低速后退
I1.2
P
MOV_W
EN
ENO
1500
IN
OUT
SMW68
Q0.2
S
1
PLS
EN
ENO
0
Q0.X
网络 6 复位
检测到B边离开时，调用停止子程序，复位完成，并进行复位指示
I1.2
N
SBR_0
EN
Q0.3
S
1
网络 7 前进，点动
I0.3
P
PLS
EN
ENO
0
Q0.X
N
SBR_0
EN

图 11-30 子程序（续）

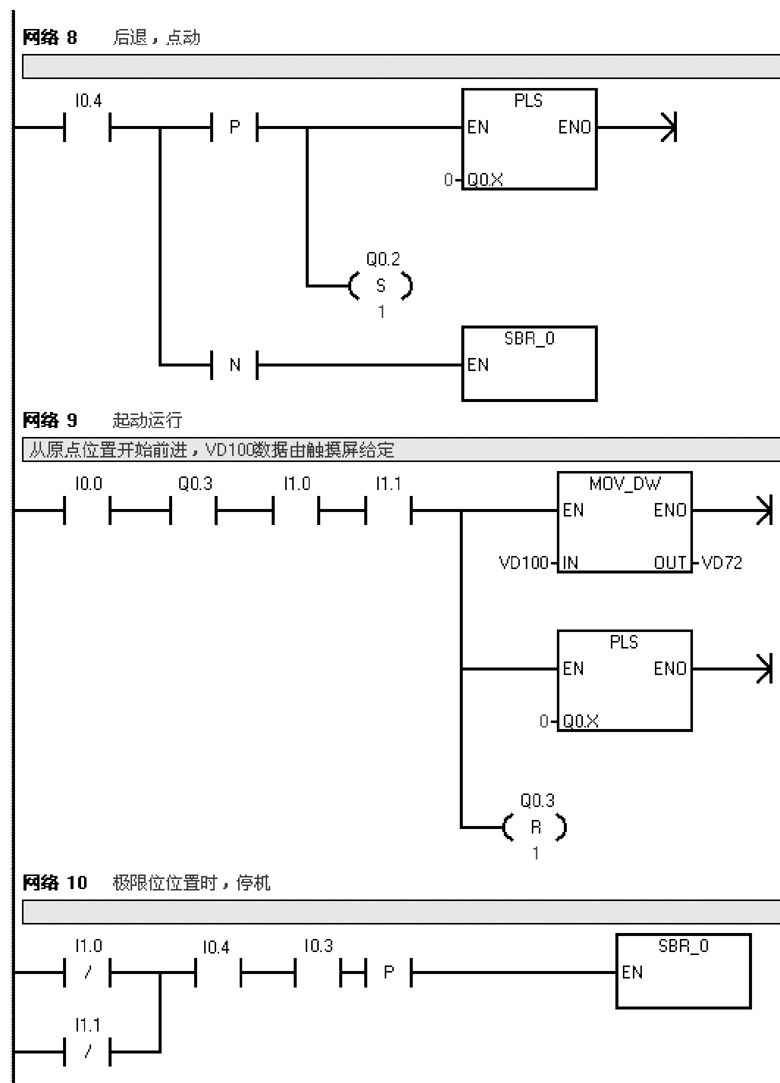


图 11-30　主程序（续）

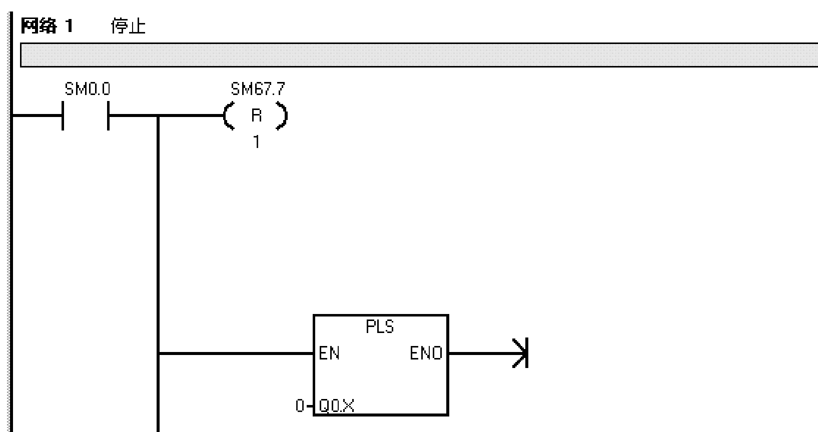


图 11-31　子程序

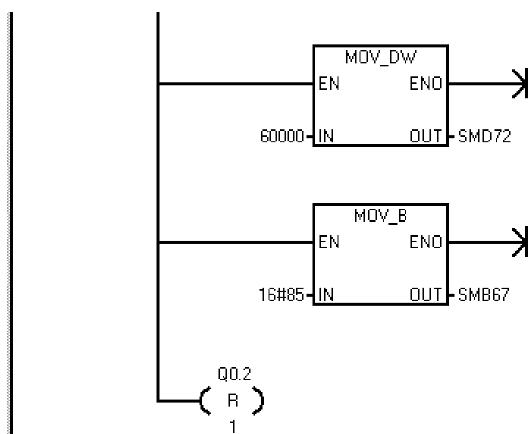


图 11-31　子程序（续）

11.9　定长前切机的 PLC 控制

【例 11-9】用 PLC 实现钢板定长剪切机的控制。系统工作过程描述如下：

薄钢板定长剪切机的输送机由伺服系统控制输送长度，剪切机采用切刀切割。初始位置时，切刀在上方（即气缸活塞位于上极限位置），当输送机输送指定的长度后，切刀向下运动，完成切割并延时 0.5 s 后，回到初始位置，完成一个工作循环。薄钢板定长剪切机的结构如图 11-32 所示。

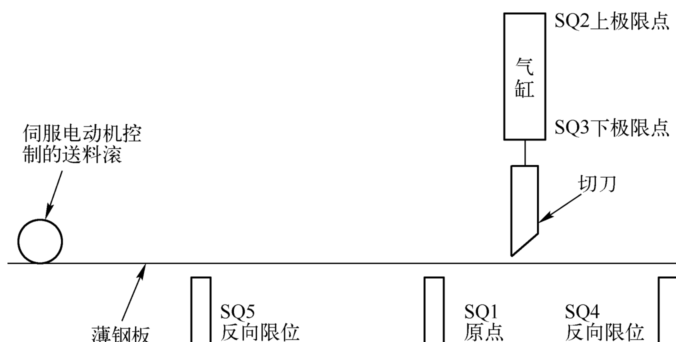


图 11-32　薄钢板定长剪切机的工作示意图

其控制任务要求如下：

① 有手动/自动转换开关 SA1，手动模式时，可以手动对切刀进行上行和下行控制。手动按钮分别是 SB5 和 SB6。

② 自动模式时，当切刀在初始位置，按下“起动”按钮 SB1，输送机送料→送料完成→切刀下行到下极限位置延时 0.5 s→回到初始位置，如此循环。

③ 在自动模式时，按下“复位”按钮 SB2，切刀回到初始位置，送料机构将料送到原点（SQ1），复位完成时，有指示灯闪亮。当按下“停止”按钮 SB3 时，系统完成一个循环

后，停止到初始位置，当按下“急停”按钮 SB4 时，立即停止。

④ 每完成一次自动切割循环，计数值加 1。

11.9.1 系统软硬件配置

1. 系统软硬件

① 1 套 STEP 7 - Micro/WIN V4.0 SP9。

② 1 台伺服电动机的型号为 HF - KE13W1 - S100。

③ 1 台伺服驱动器的型号为 MR - E - A。

④ 1 台 CPU 224XPsi。

2. PLC 的 I/O 分配

PLC 的 I/O 分配见表 11-12。

表 11-12 I/O 分配表

输入			输出		
名称	代号	地址	名称	代号	地址
原点	SQ1	I0.0	高速输出		Q0.0
上限位	SQ2	I0.1	伺服 ON		Q0.1
下限位	SQ3	I0.2	方向控制		Q0.2
起动按钮	SB1	I0.3	气缸下行	YV1	Q0.3
复位按钮	SB2	I0.4	气缸上行	YV2	Q0.4
停止按钮	SB3	I0.5	复位显示	HL1	Q0.5
手动/自动转换按钮	SA1	I0.6			
急停按钮	SB4	I0.7			
手动上行按钮	SB5	I1.0			
手动下行按钮	SB6	I1.1			
正向限位	SQ4	I1.2			
反向限位	SQ5	I1.3			
伺服准备好	RD	I1.4			

定长剪切机的原理图如图 11-33 所示。

【关键点】 连线时，务必注意 PLC 与伺服驱动器必须共地，否则不能形成回路；此外，三菱的伺服驱动器只能接受 NPN 信号，因此在选择 PLC 时，要注意选用 NPN 输出的 PLC，西门子的 S7 - 200 系列的 PLC 目前只有一款（CPU 224XPsi）是 NPN 输出。若读者一定要选用 PNP 输出 PLC，则需要将信号进行转换，通常处理信号比较麻烦而且效果要差一些。

3. 伺服电动机的参数设定

用 PLC 的高速输出点控制伺服电动机除了接线比用 PLC 的高速输出点控制步进电动机复杂外，后者不需要设置参数（细分的设置除外），而要伺服系统正常运行，必须对伺服系统进行必要的参数设置。参数设置如下：

① P0 = 0000，含义是位置控制，不进行再生制动。

② P3 = 100，含义是齿轮比的分子。

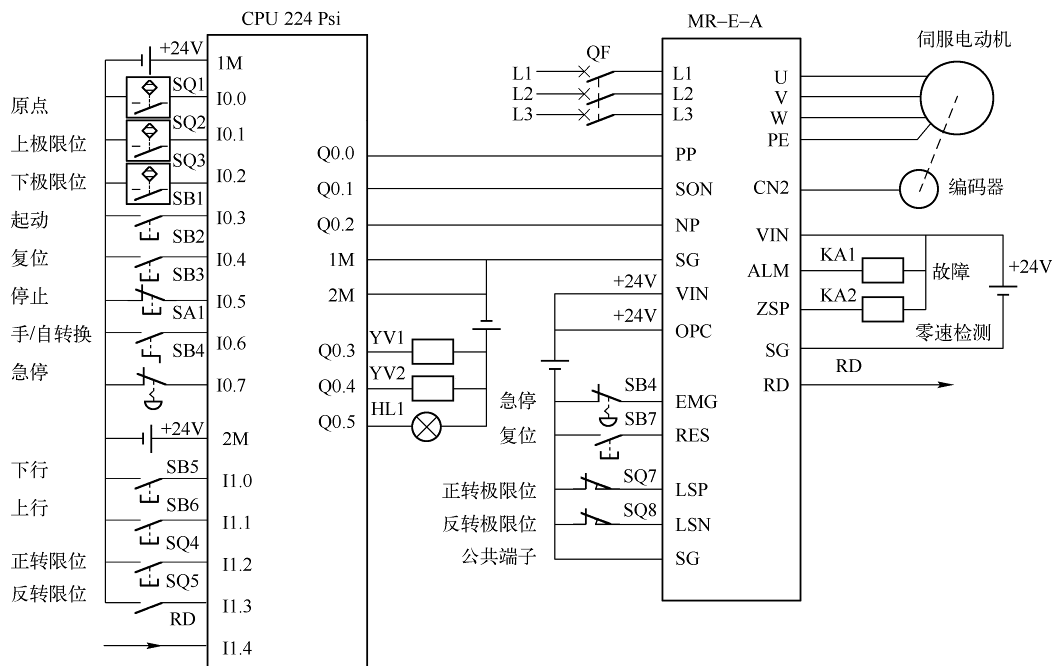


图 11-33　定长剪切机的接线图

③ P4 =1，含义是齿轮比的分母。

④ P41 =0，含义是伺服 ON、正行程限位和反行程限位都通过外部信号输入。

虽然伺服驱动器的参数很多，但对于简单的应用，只需要调整以上几个参数就足够了。

关键点：设置完成以上参数后，不要忘记保存参数，伺服驱动器断电后，以上设置才起作用。此外，有的初学者编写程序时输入的脉冲数较少，而且齿轮比 P3/P4 又很小，发现系统运行后，伺服电动机并未转动，从而不知所措，其实伺服电动机已经旋转，只不过肉眼没有发现其转动，读者只要把输入的脉冲数增加到足够大，将齿轮比调大一些，就能发现伺服电动机旋转。

11.9.2　控制程序的编写

如果用 PLS 指令编写运动控制程序，则比较麻烦，所幸西门子公司编写了运动控制指令库（MAP_SERV），使用此指令库，可以很方便实现相对运动、绝对运动、回原点和停止等功能，下面对西门子的运动指令库的指令做简要介绍，以方便读者阅读程序，需要指出，此运动指令库不是 STEP 7 - Micro/WIN 的标准配置，需要另行购买和安装。

（1）已经被指令库占用的地址

西门子运动指令库已经预定义了一些地址，见表 11-13。

表 11-13　预定义地址表

名称	MAP SERV Q0. 0	MAP SERV Q0. 1
脉冲输出	Q0. 0	Q0. 1
方向输出	Q0. 2	Q0. 3
参考点输入	I0. 0	I0. 1

（续）

名称	MAP SERV Q0.0	MAP SERV Q0.1
所用的高速计数器	HC0	HC3
高速计数器预置值	SMD 42	SMD 142
手动速度	SMD 172	SMD 182

【关键点】如果读者要使用运动指令库的“MAP SERV Q0.0”，那么 Q0.0 就作为高速输出使用，Q0.2 就作为方向信号使用，I0.0 作为参考点使用，不能另作他用。但是如果只使用了“MAP SERV Q0.0”，而没有使用“MAP SERV Q0.1”，那么 QO.1 和 Q0.3 可以由读者自行分配使用。

（2）Q0_x_CTRL 指令

该指令用于传递全局参数，每个扫描周期都需要被调用。功能描述见表 11-14。

表 11-14　Q0_x_CTRL 指令功能描述表

梯形图	参数	类型	格式	单位	意义
Q0_0_CTRL EN Velocity_SS　C_Pos Velocity_Max accel_dec_time Fwd_Limit Rev_Limit	Velocity_SS	IN	DINT	Pulse/s	起动/停止频率
	Velocity_Max	IN	DINT	Pulse/s	最大频率
	accel_dec_time	IN	REAL	s	最大加减速时间
	Fwd_Limit	IN	BOOL		正向限位开关
	Rev_Limit	IN	BOOL		反向限位开关
	C_Pos	OUT	DINT	Pulse	当前绝对位置

以下对常用的参数做详细说明：

1）Velocity_SS 是最小脉冲频率，是加速过程的起点和减速过程的终点。

2）Velocity_Max 是最大脉冲频率，受限于电动机最大频率和 PLC 的最大输出频率。在程序中若输入超出（Velocity_SS，Velocity_Max）范围的脉冲频率，将会被 Velocity_SS 或 Velocity_Max 所取代。

3）accel_dec_time 是由 Velocity_SS 加速到 Velocity_Max 所用的时间（或由 Velocity_Max 减速到 Velocity_SS 所用的时间，两者相等），范围被规定为 0.02 ～ 32.0 s，但最好不要小于 0.5 s。超出 accel_dec_time 范围的值还是可以被写入块中，但是会导致定位过程出错。

（3）Q0_x_MoveRelative 指令

该指令用于让轴按照指定的方向，以指定的速度，运动指定的相对位移。功能描述见表 11-15。

表 11-15　Q0_x_MoveRelative 指令功能描述表

梯形图	参数	类型	格式	单位	意义
Q0_0_MoveRelative EN EXECUTE Num_Pulses　Done Velocity Direction	EXECUTE	IN	BOOL		相对位移运动的执行位
	Num_Pulses	IN	DINT	Pulse	相对位移（必须 >1）
	Velocity	IN	DINT	Pulse/s	预置频率（Velocity_SS <= Velocity <= Velocity_Max）
	Direction	IN	BOOL		预置方向（0 = 反向，1 = 正向）
	Done	OUT	BOOL		完成位（1 = 完成）

（4）Scale_EU_Pulse 指令

该指令用于将一个位置量转化为一个脉冲量，因此它可用于将一段位移转化为脉冲数，或将一个速度转化为脉冲频率。功能描述见表 11-16。

表 11-16　Scale_EU_Pulse 指令功能描述表

梯　形　图	参　　数	类　　型	格　　式	单　　位	意　　义
Scale_EU_Pulse EN Input　Output Pulses E_Units	Input	IN	REAL	mm or mm/s	欲转换的位移或速度
	Pulses	IN	DINT	Pulse /revol	电动机转一圈所需要的脉冲数
	E_Units	IN	REAL	mm /revol	电动机转一圈所产生的位移
	Output	OUT	DINT	Pulse or pulse/s	转换后的脉冲数或脉冲频率

以上指令实现的功能用如下公式表示：

$$\text{Output} = \frac{\text{Pulses}}{\text{E_Units}} \cdot \text{Input}$$

（5）Q0_x_Home

Q0_x_Home 指令的功能描述见表 11-17。

表 11-17　Q0_x_Home 指令功能描述表

梯　形　图	参　　数	类　　型	格　　式	单　　位	意　　义
Q0_0_Home EN EXECUTE Position　Done Start_Dir　Error	EXECUTE	IN	BOOL		寻找参考点的执行位
	Position	IN	DINT	Pulse	参考点的绝对位移
	Start_Dir	IN	BOOL		寻找参考点的起始方向（0 = 反向，1 = 正向）
	Done	OUT	BOOL		完成位（1 = 完成）
	Error	OUT	BOOL		故障位（1 = 故障）

该指令用于寻找参考点，在寻找过程的起始，电动机首先以 Start_Dir 的方向，Homing_Fast_Spd 的速度开始寻找；在碰到限位开关（“Fwd_Limit” 或 “Rev_Limit”）后，减速至停止，然后开始相反方向的寻找；当碰到参考点开关（I0.0 或 I0.1）的上升沿时，开始减速到 “Homing_Slow_Spd”。如果此时的方向与 “Final_Dir” 相同，则在碰到参考点开关下降沿时停止运动，并且将计数器 HC0 的计数值设为 “Position” 中所定义的值。

如果当前方向与 “Final_Dir” 不同，则必然要改变运动方向，这样就可以保证参考点始终在参考点开关的同一侧（具体是哪一侧取决于 “Final_Dir”）。原点、正向限位和反向限位示意如图 11-34 所示。

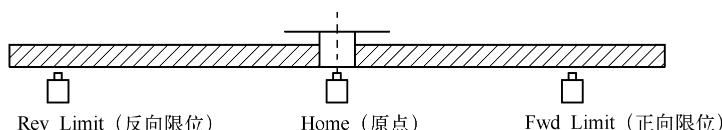


图 11-34　回参考点（原点）、正向限位、反向限位示意图

寻找参考点的状态可以通过全局变量 “Homing_State” 来监测，见表 11-18。

表 11-18　寻找参考点的状态

Homing_State 的值	意　　义
0	参考点已找到
2	开始寻找
4	在相反方向，以速度 Homing_Fast_Spd 继续寻找过程（在碰到限位开关或参考点开关之后）
6	发现参考点，开始减速过程
7	在方向 Final_Dir，以速度 Homing_Slow_Spd 继续寻找过程（在参考点已经在 Homing_Fast_Spd 的速度下被发现之后）
10	故障（在两个限位开关之间并未发现参考点）

（6）Q0_x_MoveVelocity 指令

该指令用于让轴按照指定的方向和频率运动，在运动过程中可对频率进行更改。Q0_x_MoveVelocity 指令功能描述见表 11-19。

表 11-19　Q0_x_MoveVelocity 指令功能描述表

梯　形　图	参　　数	类　　型	格　　式	单　　位	意　　义
Q0_0_MoveVelocity EN EXECUTE Velocity　Error Direction　C_Pos	EXECUTE	IN	BOOL		执行位
	Velocity	IN	DINT	Pulse/s	预置频率（Velocity_SS <= Velocity <= Velocity_Max）
	Direction	IN	BOOL		预置方向（0＝反向，1＝正向）
	Error	OUT	BYTE		故障标识（0＝无故障，1＝立即停止，3＝执行错误）
	C_Pos	OUT	DINT	Pulse	当前绝对位置

（7）Q0_x_Stop 指令

Q0_x_Stop 指令用于使轴减速直至停止。Q0_x_Stop 指令的功能描述见表 11-20。

表 11-20　Q0_x_Stop 指令功能描述表

梯　形　图	参　　数	类　　型	格　　式	意　　义
Q0_0_Stop EN EXECUTE Done	EXECUTE	IN	BOOL	执行位
	Done	OUT	BOOL	完成位（1＝完成）

【关键点】 Q0_x_MoveVelocity 功能块只能通过 Q0_x_Stop block 功能块来停止轴的运动。如图 11-35 所示。

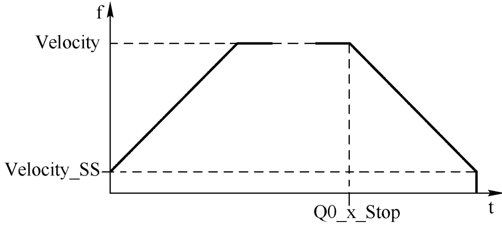


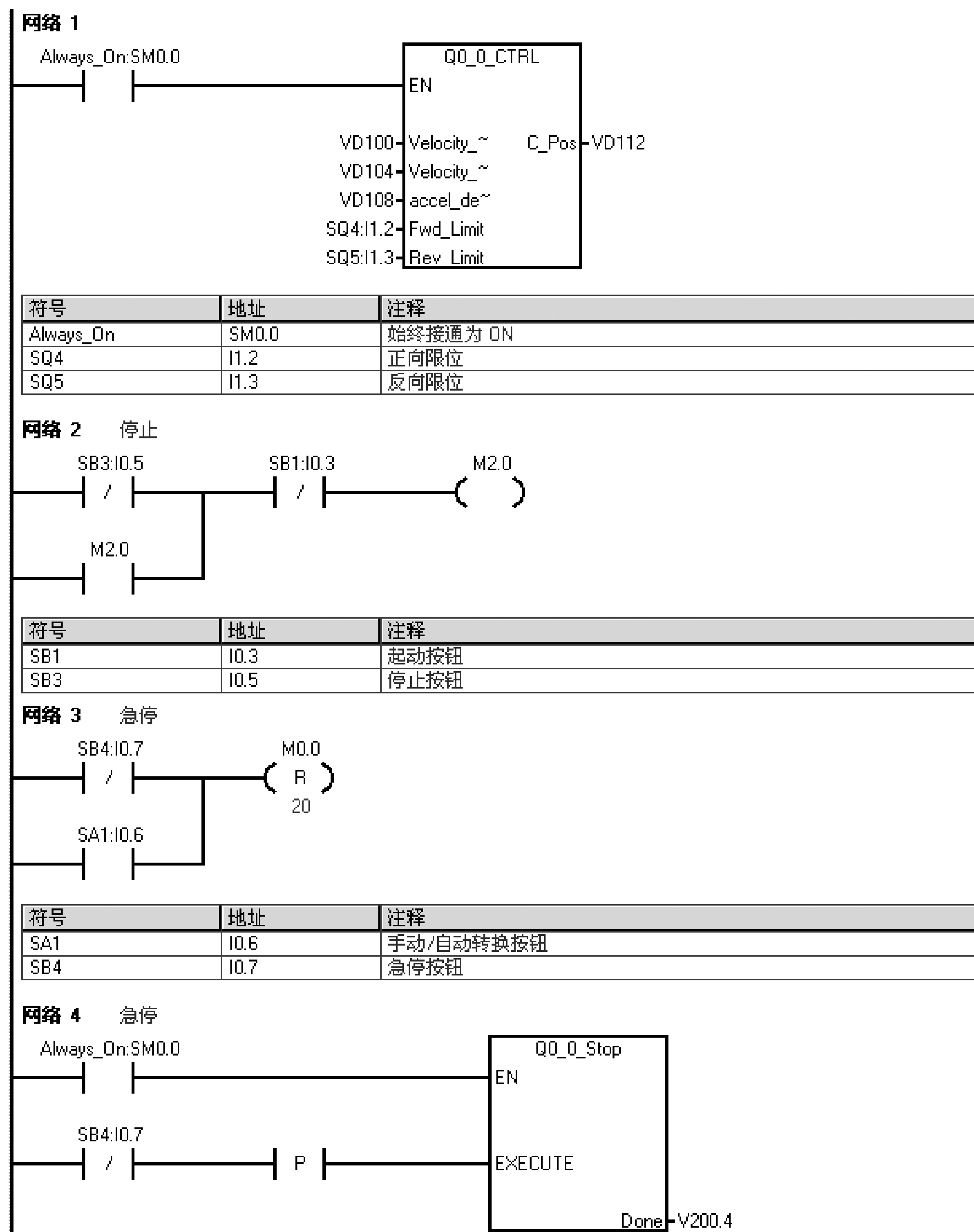
图 11-35　Q0_x_Stop 停止示意图

（8）编写程序

数据块如图 11–36 所示。主程序如图 11–37 所示。

```
VD100   10000
VD104   100000
VD108   1.0
VB200   2#1001                    //V200.0=1,V200.3=1
```

图 11–36　数据块



网络 1

符号	地址	注释
Always_On	SM0.0	始终接通为 ON
SQ4	I1.2	正向限位
SQ5	I1.3	反向限位

网络 2

符号	地址	注释
SB1	I0.3	起动按钮
SB3	I0.5	停止按钮

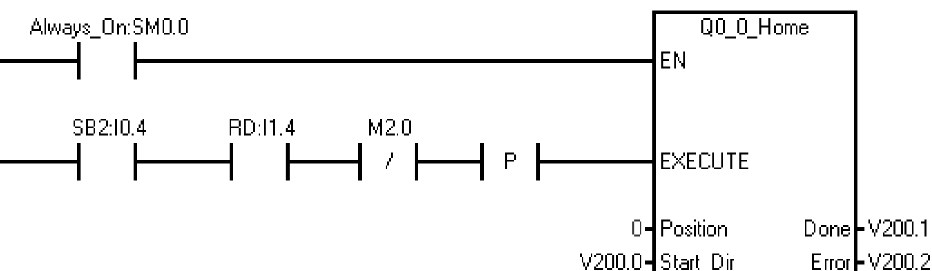
网络 3

符号	地址	注释
SA1	I0.6	手动/自动转换按钮
SB4	I0.7	急停按钮

图 11–37　主程序

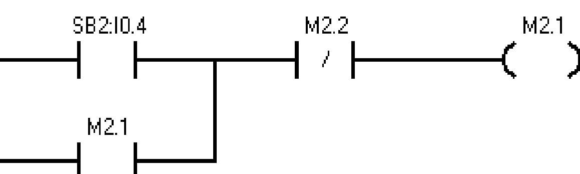
符号	地址	注释
Always_On	SM0.0	始终接通为 ON
SB4	I0.7	急停按钮

网络 5　复位：钢板的起点到原点



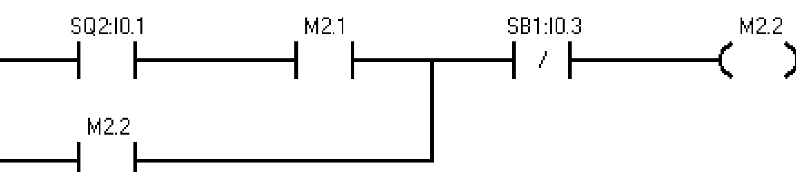
符号	地址	注释
Always_On	SM0.0	始终接通为 ON
RD	I1.4	伺服准备好
SB2	I0.4	复位按钮

网络 6　复位：气缸向上行



符号	地址	注释
SB2	I0.4	复位按钮

网络 7　复位时，气缸上行到上极限位



符号	地址	注释
SB1	I0.3	起动按钮
SQ2	I0.1	上限位

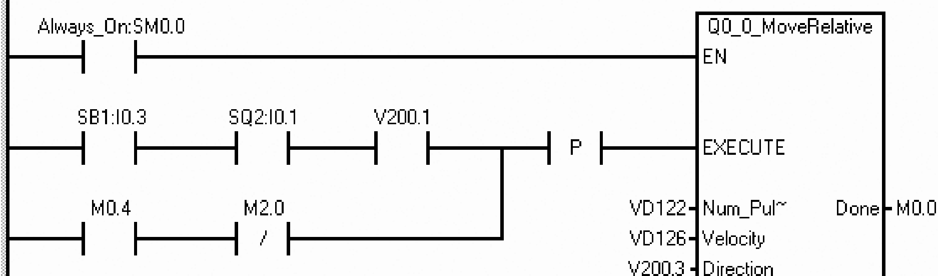
网络 8　复位完成，复位灯闪亮



符号	地址	注释
Clock_1s	SM0.5	在 1s 的循环周期内，接通为 ON 0.5s，关断为 OFF 0.5s
HL1	Q0.5	复位显示

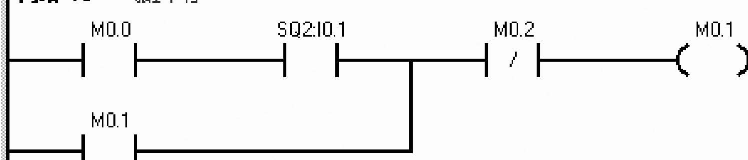
图 11-37　主程序（续）

网络 9　起动运行，钢板输送



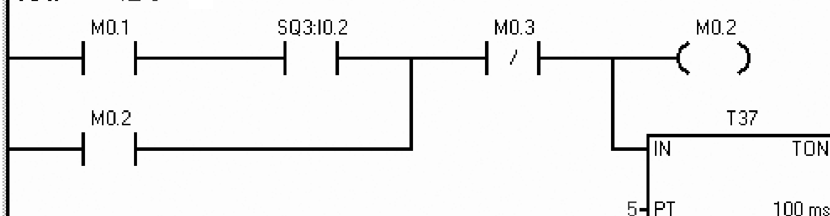
符号	地址	注释
Always_On	SM0.0	始终接通为 ON
SB1	I0.3	起动按钮
SQ2	I0.1	上限位

网络 10　气缸下行



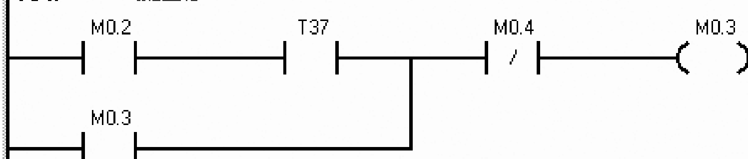
符号	地址	注释
SQ2	I0.1	上限位

网络 11　延时0.5s

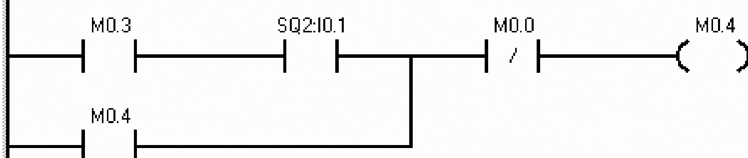


符号	地址	注释
SQ3	I0.2	下限位

网络 12　气缸上行

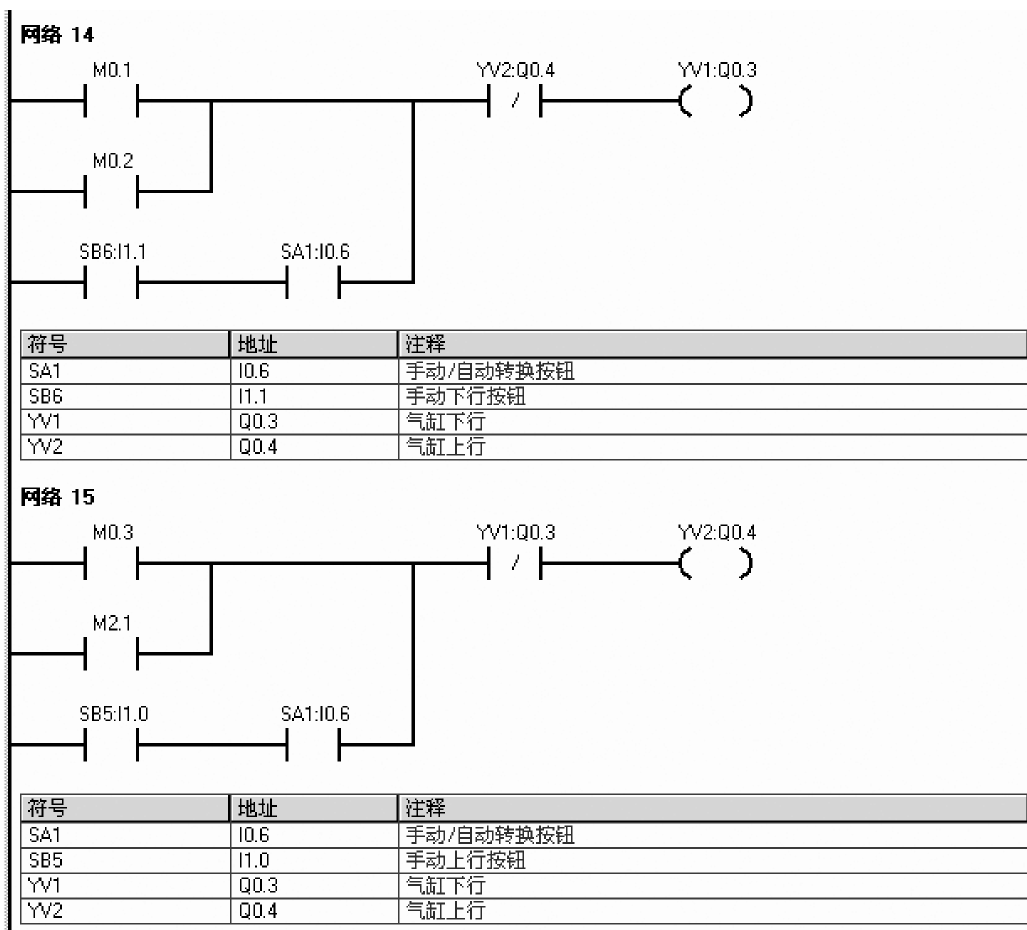


网络 13　气缸上行到位



符号	地址	注释
SQ2	I0.1	上限位

图 11-37　主程序（续）



符号	地址	注释
SA1	I0.6	手动/自动转换按钮
SB6	I1.1	手动下行按钮
YV1	Q0.3	气缸下行
YV2	Q0.4	气缸上行

符号	地址	注释
SA1	I0.6	手动/自动转换按钮
SB5	I1.0	手动上行按钮
YV1	Q0.3	气缸下行
YV2	Q0.4	气缸上行

图 11-37　主程序（续）

11.10　工业氮气管道流量监控系统的 PLC 控制

【例 11-10】某电子元件厂，因为其工艺要求，需要在包装袋中填充氮气，已知公司有三条氮气管道，管道距离在 20 m 左右，每条管道给一条生产线供气。每条氮气管道上安装有一台流量计。已知流量计的输出电流信号的范围是 0～20 mA，对应的流量是 0～200 L/min。中央控制室就在第一根管道附近。具体的技术要求如下：

1）设计一个管道的氮气流量监控系统，实时采集氮气管路上的流量信号，当管路中的流量低于 20 L/min 或者高于 160 L/min 时，监控系统发出声光报警信息。

2）三条生产线是独立运行的生产线，要求三根管道的报警系统可以独立运行，同时也能向中央控制室发送流量是否正常和实时流量数据，这些数据信息显示在人机界面上。

11.10.1　系统软硬件配置

1. 系统软硬件

① 3 台 CPU 226CN。

② 3 台 EM 231。

③ 1 套 STEP 7 - Micro/WIN V4.0 SP9。

④ 1 根 PC/PPI 电缆。

⑤ 1 台 HMI。

2. 总方案的设计

根据项目的描述，得出如下方案：

监控第一条管道的 PLC 确定为主站，设置主站的站地址为 2；监控其余两条管道的 PLC 为从站，设置其站地址分别为 3 和 4。工作站 2、3 和 4 分别监控各自管道的流量信息，工作站 3 和 4 要把实时流量信息发送到工作站 2 上，人机界面（HMI）从主站（工作站 2）上采集实时流量信息，并显示在人机界面上。总方案如图 11-38 所示。

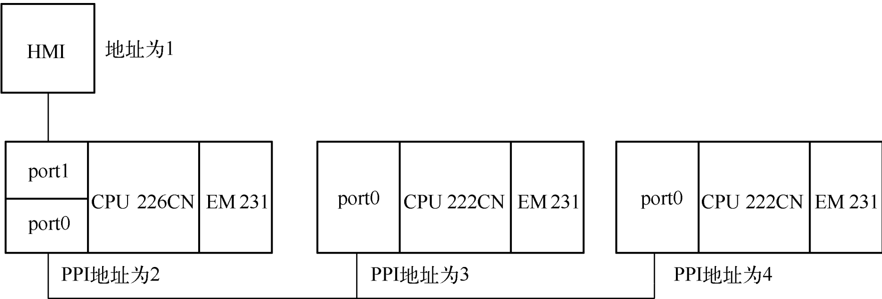


图 11-38　总方案图

3. I/O 分配

在 I/O 分配之前，先计算所需要的 I/O 点数，输出点为 2 个，不需要输入点，由于输入输出最好留 15% 左右的余量备用，初步选择从站的 PLC 是 CPU 221CN 或者 CPU 222CN，又因为要使用扩展模块 EM 231CN，所以不能选择 CPU 221CN。主站既要与从站通信，又要与 HMI 通信，所以需要 2 个通信接口，选定为 CPU 226CN。西门子 S7 - 200 系列的 PLC，继电器输出的类型要比晶体管输出类型的 PLC 驱动能力要大得多，所以 PLC 最后定为 CPU 222CN（AC/DC/继电器）。氮气管路流量监控系统的主站的 I/O 分配见表 11-21。从站的 I/O 分配也是一样，在此不再重复。

表 11-21　I/O 分配表

输　　入			输　　出		
名 称	符 号	输入点	名 称	符 号	输出点
模拟量输入		AIW0	报警灯	HL	Q0.0
			报警铃	BL	Q0.1

【关键点】 主站的 CPU 也可以选择 CPU 224XP，但价格稍贵；还可以选择 CPU 222 和 EM 277 模块组合，但必须使用西门子的 HMI。

4. 设计电气原理图

根据 I/O 分配表和题意，设计主站的原理图如图 11-39 所示，从站的原理图与之类似，

只要把 CPU 226CN 改为 CPU 222CN 即可，其他部分相同，在此不再重复。

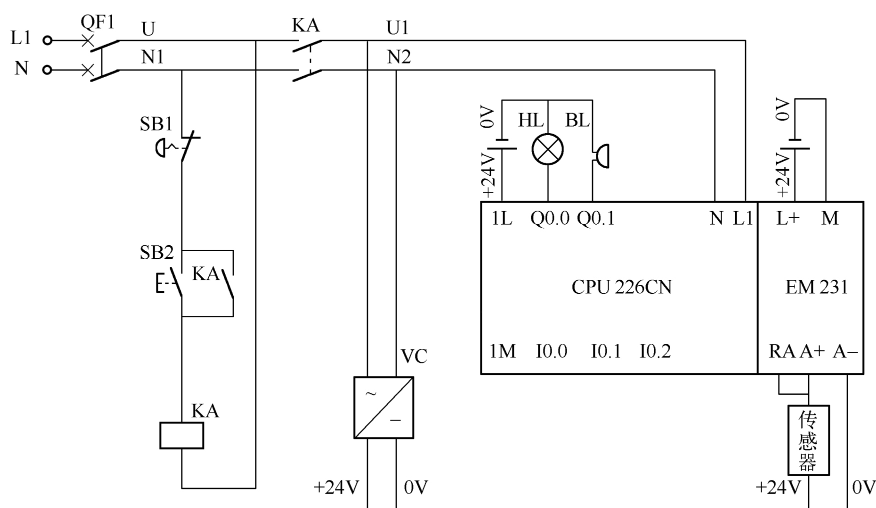


图 11-39　主站的原理图

11.10.2　编写控制程序

（1）列出数据缓冲区

主站发送数据缓冲区和从站接收数据缓冲区分别见表 11-22 和表 11-23。

表 11-22　主站发送缓冲区

VB200	状　态　字
VB201	从站的地址（3）
VD202	&VB200 从站的接收缓冲区地址
VB206	2（字节）
VB207	VW207
VB208	

⇦

表 11-23　从站接收缓冲区

	从站流量
VB200	VW200
VB201	

主站接收数据缓冲区和从站发送数据缓冲区分别见表 11-24 和表 11-25。

表 11-24　主站接收缓冲区

VB300	状　态　字
VB301	从站的地址（4）
VD302	&VB300 从站的接收缓冲区地址
VB306	2（字节）
VB307	VW307
VB308	

⇦

表 11-25　从站发送缓冲区

	从站流量
VB300	VW300
VB301	

（2）编写程序

程序如图 11-40、图 11-41、图 11-42 和图 11-43 所示。

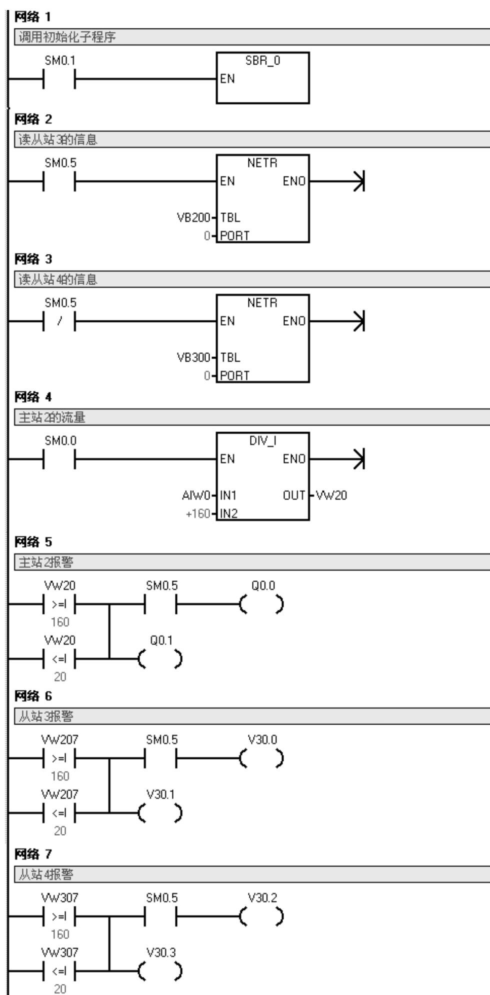


图 11-40　主站 2 主程序

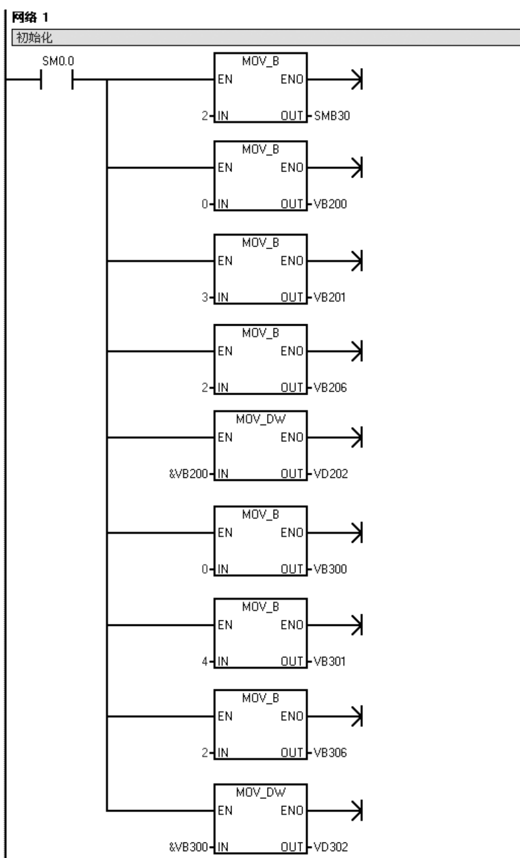


图 11-41　主站 2 子程序

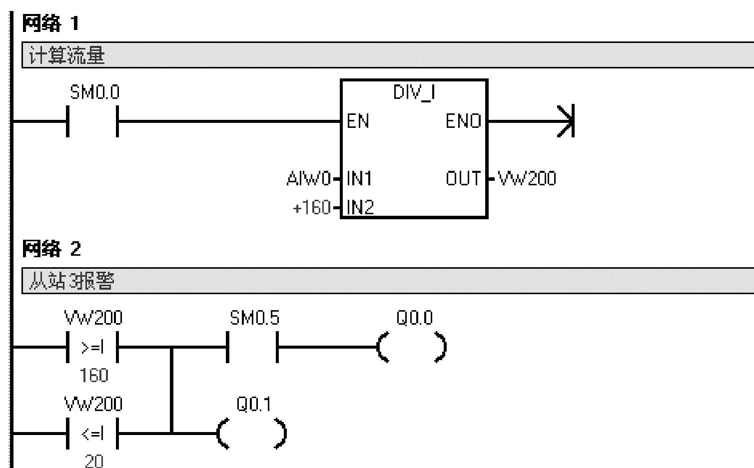


图 11-42　从站 3 程序

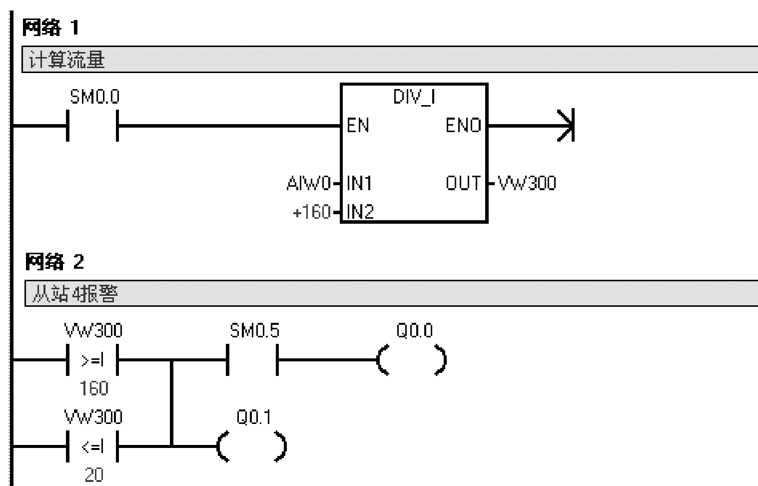


图 11-43　从站 4 程序

11.11　物料搅拌机的 PLC 控制

【例 11-11】有一个物料搅拌机，主机由 7.5kW 的电动机带动，根据物料不同，要求速度在一定的范围内无极可调，且要求物料太多或者卡死设备时系统能及时保护；机器上配有冷却水，冷却水温度不能超过 50℃，而且冷却水管不能堵塞，也不能缺水，堵塞和缺水将造成严重后果，冷却水的动力不在本设备上，水温和压力要可以显示。

11.11.1　系统软硬件配置

1. 系统的软硬件

① 1 台 CPU 226CN。

② 1 台 EM 231。

③ 1 台 TP－177B。

④ 1 台 MM 440。

⑤ 1 台压力传感器（含变送器）。

⑥ 1 台温度传感器（含变送器）。

⑦ 1 套 STEP 7－Micro/WIN V4 SP9。

⑧ 1 套 WINCC FLEXIBLE 2008 SP4。

2. PLC 的 I/O 分配

PLC 的 I/O 分配见表 11-26。

表 11-26　PLC 的 I/O 分配表

序　　号	地　　址	功　　能	序　　号	地　　址	功　　能
1	I0.0	起动	9	AIW2	压力
2	I0.1	停止	10	VD0	速度
3	I0.2	急停	11	VD22	电流值
4	M0.0	起/停	12	VD50	转速设定

（续）

序　　号	地　　址	功　　能	序　　号	地　　址	功　　能
5	M0. 1	缓停	13	VW60	温度显示
6	M0. 2	起/停	14	VW62	压力显示
7	M0. 3	快速停			
8	AIW0	温度			

3. 原理图

系统的原理图如图 11-44 所示。

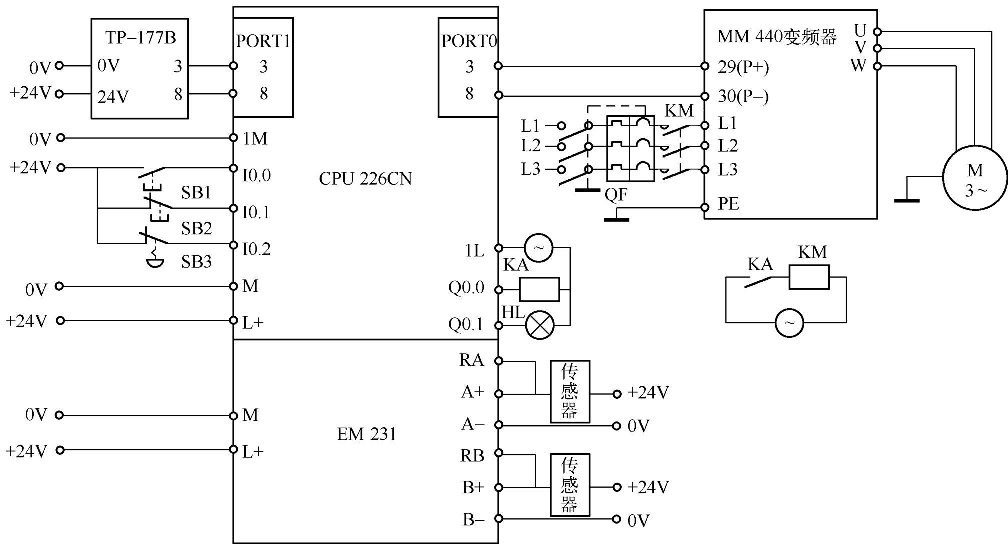


图 11-44　原理图

4. 变频器参数设定

变频器的参数设置见表 11-27。

表 11-27　变频器的参数设置

序　　号	变频器参数	出　厂　值	设　定　值	功能说明
1	P0005	21	27	显示电流值
2	P0304	230	380	电动机的额定电压（380 V）
3	P0305	3. 25	0. 35	电动机的额定电流（0. 35 A）
4	P0307	0. 75	0. 06	电动机的额定功率（60 W）
5	P0310	50	50	电动机的额定频率（50 Hz）
6	P0311	0	1 440	电动机的额定转速（1 430 r/min）
7	P0700	2	5	选择命令源（COM 链路的 USS 设置）
8	P1000	2	5	频率源（COM 链路的 USS 设置）
9	P2010	6	6	USS 波特率（6～9 600）
10	P2011	0	18	站点的地址
11	P2012	2	2	PZD 长度
12	P2013	127	127	PKW 长度（长度可变）
13	P2014	0	0	看门狗时间

11.11.2　编写控制程序

温度传感器最大测量量程是 100℃，其对应的数字量是 32 000，所以 AIW0 采集的数字量除 32 000 再乘 100（即 AIW0/320）就是温度值；压力传感器的最大量程是 10 000 Pa，其对应的数字量是 32 000，所以 AIW2 采集的数字量除 32 000 再乘 10 000（即 AIW2×10/32）就是压力值；程序中的 VD0 是额定频率的百分比，由于电动机的额定转速是 1 400 r/min，假设电动机转速是 700 r/min，那么 VD0 = 50.0，所以 VD0 = VD50/1400×100×（VD50/140）。

程序如图 11-45 所示。

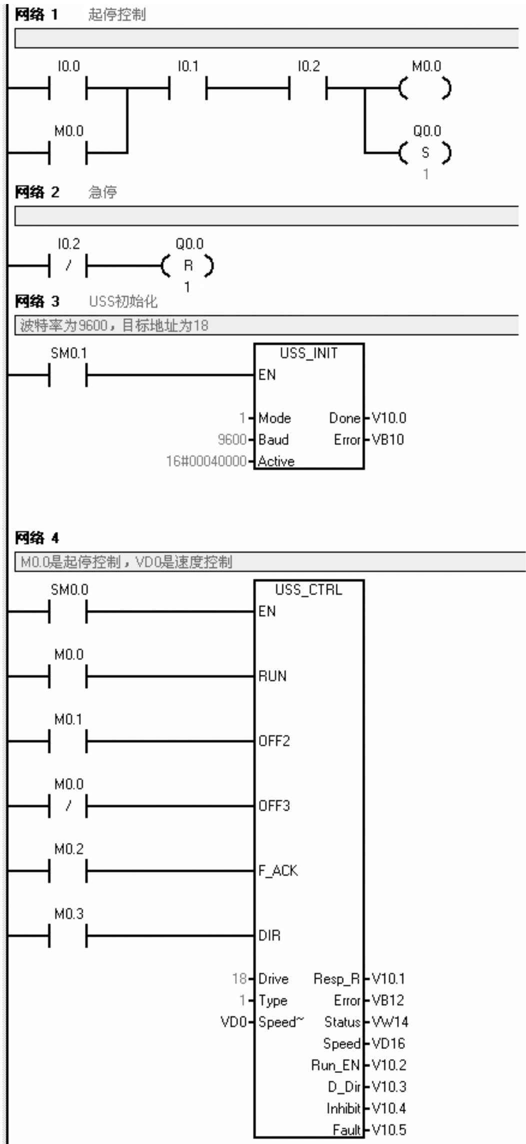


图 11-45　程序

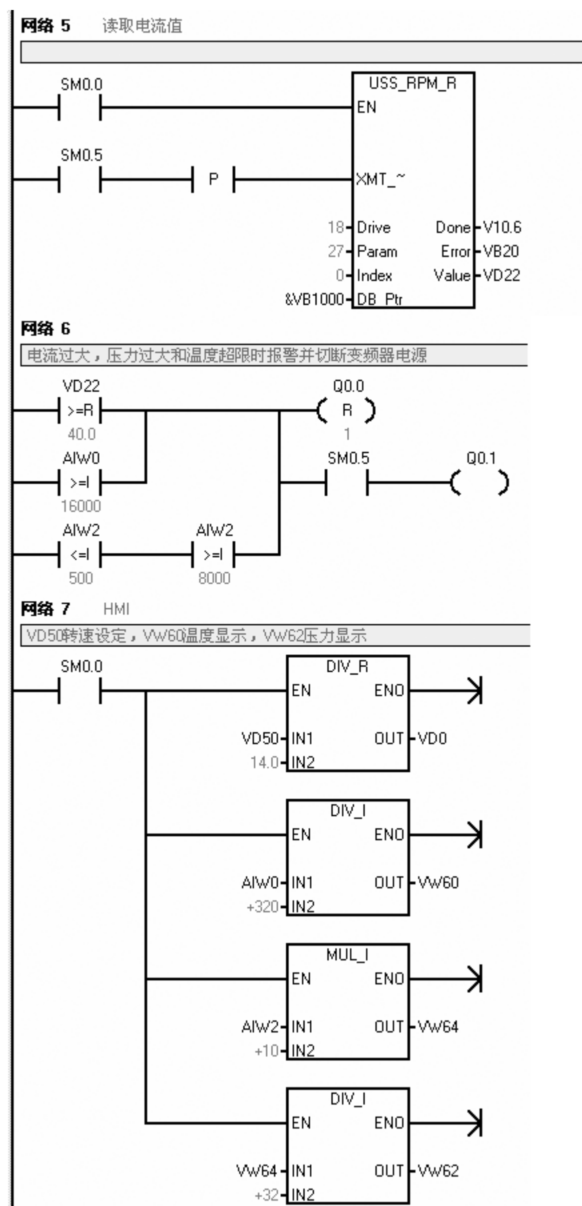


图 11-45 程序（续）

注意：压力和温度显示都是整数，若要显示为实数，程序要稍加改动；此外，关于触摸屏部分在此不作说明，请读者自行完成。

小 结

（1）本章的 1 ~4 共 4 个例子，是逻辑控制的应用，相对比较容易，但这是学习和应用 PLC 的基本，读者必须掌握。

（2）本章的第 5 和 6 个例子，是 PLC 和变频器的系统集成，前者是涉及模拟量速度给

定，后者涉及通信速度给定，有一定的难度。

（3）本章的第 5 和 7 个例子，是 PLC 的逻辑控制和高速输入的应用，有一定的难度。

（4）本章的第 8 和 9 个例子，是 PLC 的高速输出的应用，还涉及步进驱动系统和伺服驱动系统，这是最有难度的部分。

（5）本章的第 10 个例子，是 PLC 的 PPI 的应用，通信是 PLC 应用的难点。

（6）本章的第 11 个例子，是 PLC 的通信、变频器和 HMI 的应用，这是一个典型工程应用。

习　　题

1. 深孔钻组合机床的 PLC 控制。深孔钻加工时，为了排屑和散热，需要周期性地从工件中退出钻头，刀具的进退与行程开关如图 11-46 所示。在起始位置 O 时，SQ1 限位开关被压住，按下起动按钮时，电动机正转起动，刀具前进；刀具的后退由限位开关控制，当碰到 SQ2、SQ3、SQ4 电动机反转后退，当一个循环完成后，机床停机。当按下停止按钮 SB2 时，机床停机。机床还有前进点动按钮 SB3 和后退点动按钮 SB4。

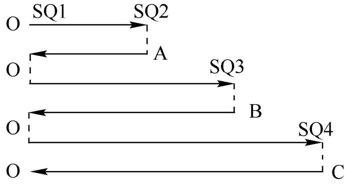


图 11-46　示意图

2. 一组皮带由三条皮带连接而成，用于传送一定长度的金属板，为了避免传送带在没有物料时空转，每个传送带的末端有一个接近开关用于检测金属板。传送带检测到金属板才能起动，当金属板离开传送带时停止。示意图如图 11-47 所示。

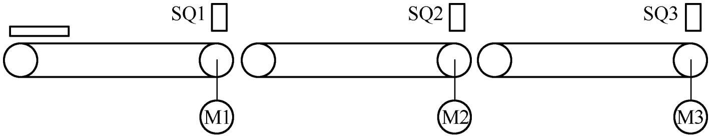


图 11-47　示意图

参 考 文 献

［1］向晓汉，等．PLC 控制技术与应用［M］．北京：清华大学出版社．2010.

［2］廖常初．PLC 编程及应用［M］．北京：机械工业出版社．2008.

［3］西门子（中国）有限公司．S7－200 可编程控制器系统手册．2009.

［4］西门子（中国）有限公司．MICROMASTER 440 标准变频器使用大全．2011.

［5］蔡行健．深入浅出西门子 S7－200 PLC［M］．北京：北京航空航天大学出版社．2003.

［6］张胜宇．可编程控制器实训项目式教程［M］．北京：电子工业出版社．2012.

［7］张云刚．从入门到精通—S7－300/400 PLC 技术与应用［M］．北京：人民邮电出版社．2007.



本科电气精品教材推荐

西门子工业自动化系列教材

西门子 S7-300/400PLC 编程与应用

书号：28666　　定价：43.00 元

作者：刘华波　　配套资源：DVD 光盘

推荐简言：

本书由浅入深全面介绍了西门子公司广泛应用的大中型 PLC——S7-300/400 的编程与应用，注重示例，强调应用。全书共分为 14 章，分别介绍了 S7 系统概述，硬件安装与维护，编程基础，基本指令，符号功能，测试功能，数据块，结构化编程，模拟量处理与闭环控制，组织块，故障诊断，通信网络等。

西门子人机界面（触摸屏）组态与应用技术 第 2 版

书号：19896　　定价：40.00 元

作者：廖常初　　配套资源：DVD 光盘

推荐简言：本书介绍了人机界面与触摸屏的工作原理和应用技术，通过大量的实例，深入浅出地介绍了使用组态软件 WinCC flexible 对西门子的人机界面进行组态和模拟调试的方法，以及文本显示器 TD200 的使用方法。介绍了在控制系统中应用人机界面的工程实例和用 WinCC flexible 对人机界面的运行进行离线模拟和在线模拟的方法。随书光盘提供了大量西门子人机界面产品和组态软件的用户手册，还提供了作者编写的与教材配套的例程，读者用例程在计算机上做模拟实验。

西门子 S7-200PLC 工程应用技术教程

书号：31097　　定价：55.00 元

作者：姜建芳　　配套资源：DVD 光盘

推荐简言：

本书以西门子 S7-200 PLC 为教学目标机，在讨论 PLC 理论基础上，注重理论与工程实践相结合，把 PLC 控制系统工程设计思想和方法及其工程实例融合到本书的讨论内容中，使本书具有了工程性与系统性等特点。便于读者在学习过程中理论联系实际，较好地掌握 PLC 理论基础知识和工程应用技术。

西门子 S7-1200 PLC 编程与应用

书号：34922　　定价：42.00 元

作者：刘华波　　配套资源：DVD 光盘

推荐简言：

本书全面介绍了西门子公司新推出的 S7-1200 PLC 的编程与应用。全书共分为 9 章，分别介绍了 PLC 的基础知识、硬件安装与维护、编程基础、基本指令、程序设计、结构化编程、精简面板组态、通信网络、工艺功能等。

工业自动化技术

书号：35042　　定价：39.00 元

作者：陈瑞阳　　配套资源：DVD 光盘

推荐简言：本书内容涵盖了工业自动化的核心技术，即可编程序控制器技术、现场总线网络通信技术和人机界面监控技术。在编写形式上，将理论讲授与解决生产实际问题相联系，书中以自动化工程项目设计为依托，采用项目驱动式教学模式，按照项目设计的流程，详细阐述了 PLC 硬件选型与组态、程序设计与调试、网络配置与通信、HMI 组态与设计以及故障诊断的方法。

西门子 S7-300/400PLC 编程技术及工程应用

书号：36617　　定价：38.00 元

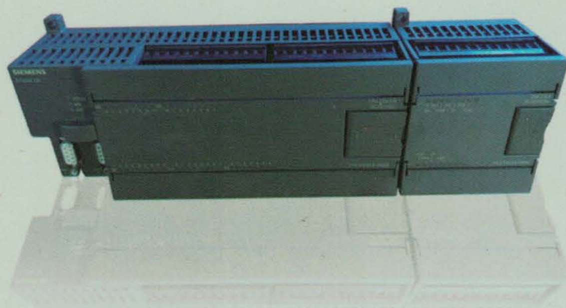
作者：陈海霞　　配套资源：电子教案

推荐简言：本书主要讲述 S7-300/400 的系统概述及 STEP 7 的使用基础；介绍了基于 IEC61131-1 的编程语言及先进的编程技术思想、组织块和系统功能块的作用、西门子通讯的种类及实现方法、工程设计步骤和工程实例。通过大量的实验案例和真实的工程实例使学习和实践能融会贯通，通过实用编程技术的介绍，提供易于交流的平台和清晰的编程思路。随书光盘内容包括书中实例和课件。

在线互动交流平台

官方微博：http://weibo.com/cmpjsj
官方博客：http://blog.sina.com.cn/cmpbookjsj
读者信箱：cmp_itbook@163.com

S7-200 PLC
基础及工程应用



本书特色

- 本书以S7-200 PLC为研究对象，从基础入门开始讲解
- 本书用150个基础案例，由浅入深、循序渐进地引导读者学习
- 本书立足工程应用，用20个工程实例，深入剖析S7-200 PLC在逻辑控制、过程控制、运动控制和通信技术中的应用
- 本书作者多来自于工程一线，所有工程实例均为作者团队编写

地址：北京市百万庄大街22号
邮政编码：100037
电话服务
社服务中心：010-88361066
销售一部：010-68326294
销售二部：010-88379649
读者购书热线：010-88379203
网络服务
教材网：http://www.cmpedu.com
机工官网：http://www.cmpbook.com
机工官博：http://weibo.com/cmp1952
封面无防伪标均为盗版

上架指导 工业技术/PLC
ISBN 978-7-111-47553-8
策划编辑◎ 时 静 / 封面设计◎



扫一扫，或搜索微信号cmpbook-IT，关注即可获得更多机工IT图书服务。

ISBN 978-7-111-47553-8



定价：69.00元